

PERISIAN PENGANALISA PLAGIARISME (Plagiarizer 2003)

Oleh:

Mohd Hairul Nizam B. Md. Nasir

WEK000349

Sessi 2002/2003

Perpustakaan SKTM

Tesis Dihantar Kepada Fakulti Sains Komputer dan Teknologi Maklumat Universiti
Malaya Sebagai Sebahagian dalam Memenuhi Keperluan Syarat Pengijazahan Sarjana
Muda Sains Komputer Dan Teknologi Maklumat



Jabatan Kejuruteraan Perisian
Fakulti Sains Komputer dan Teknologi Maklumat
Universiti Malaya
Kuala Lumpur

ABSTRAK

Perkembangan teknologi bidang perisian komputer dan penggunaannya dalam institusi pengajian menyaksikan lebih ramai para pelajar menggunakan komputer dalam menyiapkan tugas yang diberikan oleh pensyarah. Maka, gejala plagiarisme di kalangan para pelajar semakin bertambah mudah dan cepat akibat daripada penggunaan teknologi ini.

Perisian penganalisa plagiarisme ini merupakan satu perisian yang dibangunkan untuk membantu para staf akademik khususnya para pensyarah dan tutor di institusi-institusi pengajian untuk mengesan gejala plagiarisme yang berlaku di kalangan tugas para pelajar. Objektif utama pembangunan adalah untuk menjadi alat automasi bagi memudahkan para pensyarah atau tutor untuk mengesan gejala plagiarisme yang berlaku dalam set tugas pelajar di mana ia membenarkan proses analisa dilakukan dalam kuantiti yang banyak. Perisian ini berpegangan kepada 3 konsep utama iaitu ketepatan, mudah untuk digunakan dan berprestasi tinggi.

Perisian yang bakal dibangunkan ini berkebolehan untuk menjana laporan bagi melaporkan analisa yang telah dilakukan oleh sistem perisian dengan menunjukkan peratusan plagiarisme dan juga turut menunjukkan perenggan dan ayat yang terlibat dengan gejala plagiarisme ini. Maka berdasarkan laporan yang dijana ini, para pensyarah atau tutor dapat mengesan dan menentukan samada tugas pelajar tersebut terlibat dengan plagiarisme atau tidak.

Segala spesifikasi cadangan pembangunan perisian penganalisa plagiarisme ini akan diterangkan dengan lebih lanjut di dalam laporan ini. Diharapkan agar

perisian yang bakal dibangunkan ini, dapat memberi manfaat sepenuhnya terutamanya kepada institusi-institusi pengajian supaya gejala plagiarisme ini dapat dikurangkan dari semasa ke semasa seterusnya dapat melahirkan para pelajar yang dinamik, wibawa dan berpendirian.

Terima kasih kepada semua pihak yang terlibat.

Sungguh-dagat penghargaan saya ditujukan kepada penyelia projek ilmiah saya, Profesor Madya Raja Ahmad Zuhairi Hj. Raja Zaidul Kadir, dengan meluangkan masa keluarga beliau untuk berjumpa-sama saya mengenai perancangan mengenai projek ini. Sebagai cadangan, arahan, bimbingan, dan dorongan yang diberikan tidak akan saya lupakan dan ia merupakan faktor yang sangat penting bagi saya untuk menyelesaikan projek ini. Kekasih saya, isteri dan semua family kepada Profesor Madya Dr. Othman Hock untuk telah memberi bimbingan kepada saya yang telah banyak membuat cadangan, arahan, dan bimbingan untuk meningkatkan kualiti kajian yang dijalankan termasuklah bagi algoritma, perisian-kelompok bagi meningkatkan kualiti perisian, perisian-penerjemah dan berkesan.

Kerabat saya, Mohd Rizal Kadir yang telah bersedia membantu projek ini, semua family terutama ibu saya sebagai tenaga yang diberikan dan bertanggungjawab untuk membantu idea dan perisian untuk meningkatkan kualiti yang meningkatkan isi dan semasa ini secara. Terima kasih juga kepada para responden yang telah membantu saya dalam dan para pakar yang kesediaan untuk membantu saya dalam meningkatkan kualiti perisian dan perisian yang telah dibangunkan.

Selain itu, saya juga ingin mengucapkan terima kasih kepada semua pihak yang telah membantu saya dalam meningkatkan kualiti perisian dan perisian yang telah dibangunkan.

PENGHARGAAN

Bersyukur saya ke hadrat Illahi kerana dengan limpah kurnia-Nya yang telah memberi saya semangat dan kekuatan untuk menyiapkan projek ilmiah tahap akhir fasa I dalam jangka masa yang ditetapkan.

Setinggi-tinggi penghargaan saya ucapkan kepada penyelia projek ilmiah saya, Profesor Madya Raja Ainon Zabariah Bt. Raja Zainal kerana sanggup meluangkan masa berharga beliau untuk bersama-sama saya membuat perbincangan mengenai projek ini. Segala cadangan, nasihat, bimbingan dan tunjuk ajar yang diberikan tidak akan saya lupakan dan ia merupakan suatu yang amat berharga buat saya untuk menyiapkan projek ini. Rakaman penghargaan dan terima kasih kepada Profesor Madya Dr. Ow Siew Hock selaku moderator bagi projek ilmiah saya yang telah banyak memberi cadangan, nasihat dan idea untuk memperbaiki sistem perisian yang dicadangkan terutamanya dari segi algoritma penyelesaiannya bagi menghasilkan suatu perisian yang menepati piawai dan berkualiti.

Kepada rakan saya, Mohd Rizal Kadis yang turut sama-sama membuat projek ini, terima kasih diucapkan atas segala kerjasama yang diberikan dan kesanggupan untuk berkongsi idea dan pendapat untuk memperbaiki sistem yang dicadangkan ini dari semasa ke semasa. Terima kasih juga kepada para responden terutamanya para pensyarah dan para pelajar atas kesudian meluangkan masa serta memberi kerjasama semasa temubual dan soal selidik dijalankan.

Salam kasih juga buat bonda dan keluarga yang sentiasa mendoakan kejayaan anakanda. Tidak lupa juga kepada rakan-rakan yang memahami dan sentiasa berdiri

teguh mendorong kejayaan saya. Akhir kata, ribuan terima kasih saya kepada semua individu yang terlibat secara langsung atau tidak langsung dalam menjayakan latihan ilmiah ini.

KANDUNGAN

MUKASURAH

Sekian terima kasih. Wassalam.

Abstrak

ii

Penghargaan

iii

MOHD HAIRUL NIZAM B. MD. NASIR

Fakulti Sains Komputer & Teknologi Maklumat

Universiti Malaya

2002/2003

BAB 1: PENGENALAN

1.1 Definisi Projek

1

1.2 Motivasi Projek

2

1.3 Objektif Projek

4

1.4 Struktur Projek

5

1.1.1 Skop Tanggungjawab

3

1.1.2 Skop Perisian

3

1.5 Perancangan Aktiviti Projek

6

1.6 Hasil Projek

7

1.7 Organisasi Bab

8

ISI KANDUNGAN

BAB 21 - KAJIAN LITERASI

KANDUNGAN

MUKASURAT

Abstrak	ii
Penghargaan	iv
Isi Kandungan	vi
Senarai Jadual	xii
Senarai Rajah	xiv

BAB 1 : PENGENALAN

1.1	Definisi Projek	1
1.2	Motivasi Projek	2
1.3	Objektif Projek	4
1.4	Skop Projek	5
1.4.1	Skop Pengguna	5
1.4.2	Skop Perisian	5
1.5	Perancangan Aktiviti Projek	6
1.6	Hasil Jangkaan	8
1.7	Organisasi Bab	9

BAB 2 : KAJIAN LITERASI

2.1	Pengenalan	11
2.2	Plagiarisme	12
2.2.1	Definisi	12
2.2.2	Mengapa Plagiarisme berlaku	14
2.2.3	Jenis-Jenis Plagiarisme	15
2.2.4	Mengenalpasti Plagiarisme dalam Tugas Teks	17
2.2.5	Teknik-Teknik Plagiarisme yang Dilakukan	19
2.3	Produk Perisian yang Terdapat di Pasaran	20
2.3.1	Produk Perisian yang Dipilih untuk Kajian Kes	20
2.3.2	Analisis & Kesimpulan ke Atas Perisian yang Terdapat di Pasaran	29
2.4	Kajian Bahasa Pengaturcaraan	31
2.4.1	Java	31
2.4.2	C++	36
2.4.3	Kesimpulan	39
4.2	Perancangan	41
4.2.1	Kajian Penyelidikan	43
4.2.2	Definisi	44
4.3	Mod Tindakan	45
4.3.1	Sua Sebaik Pengguna Ti	47
4.3.1.1	Sebab & Akibat	48
4.3.1.2	Sebab & Akibat Pada Pelajar	49

BAB 3 : METODOLOGI PEMBANGUNAN

3.1	Pengenalan	41
3.2	Metodologi Pembangunan Sistem	42
3.3	Kajian Pemodelan Proses	43
3.3.1	Model Air Terjun	43
3.3.2	Model Prototaip	47
3.4	Pemilihan Pemodelan Proses	50
3.4.1	Justifikasi Pemodelan Proses	50
3.5	Fasa-Fasa Pembangunan Perisian	53

BAB 4 : ANALISA KEPERLUAN SISTEM

4.1	Pengenalan	59
4.2	Pengumpulan Fakta & Maklumat	59
4.2.1	Soal Selidik	60
4.2.2	Temubual	61
4.2.3	Layaran Internet	62
4.2.4	Pertanyaan	63
4.2.5	Kajian Penyelidikan	63
4.2.6	Perbincangan	64
4.3	Hasil Tinjauan	65
4.3.1	Soal Selidik Pengguna	65
4.3.1.1	Seksyen Umum	65
4.3.1.2	Seksyen Tugasan Para Pelajar	66

4.3.1.3	Seksyen Jangkaan Fungsian Perisian	68
4.3.2	Soal Selidik Pelajar	71
4.3.3	Temubual Kepada Pelajar	73
4.4	Rumusan Daripada Hasil Tinjauan	74
4.5	Keperluan Sistem	76
4.5.1	Keperluan Fungsian	76
4.5.2	Algoritma Pengiraan	84
4.5.2.1	Mengira Peratusan Plagiarisme Melalui Kaedah Peratusan Frekuensi Kesamaan Bilangan Perkataan Antara Dua Tugasan	84
4.5.2.2	Implementasi Algoritma Dalam Dunia Sebenar	87
4.5.2.3	Menentukan Tahap Peratusan Plagiarisme	89
4.5.3	Keperluan Bukan Fungsian	91
4.6	Keperluan Perisian	94
4.7	Keperluan Perkakasan	97

BAB 5 : REKABENTUK SISTEM

5.1	Pengenalan	98
5.2	Rekabentuk Logikal Sistem Perisian	100
5.2.1	Carta Alir	100
5.3	Rekabentuk Antaramuka	105
5.4	Kesimpulan	113

BAB 6 : PENGKODAN DAN PERLAKSANAAN

6.1	Pengenalan	114
6.2	Persekitaran Pembangunan	114
6.3	Fasa Pengkodan Plagiarizer 2003	115
6.3.1	Pendekatan Pengaturcaraan	119
6.4	Fasa Pelaksanaan Plagiarizer 2003	122

BAB 7 : PENGUJIAN PERISIAN

7.1	Pengenalan	130
7.2	Pengujian Perisian Plagiarizer 2003	131
7.2.1	Pengujian Unit	132
7.2.2	Pengujian Integrasi	134
7.2.3	Pengujian Perisian	134
7.2.3.1	Pengujian Kebolehgunaan	135
7.2.3.1.1	Asas Persekitaran Pengujian Kebolehgunaan	136
7.2.3.1.2	Hasil Keputusan Pengujian Kebolehgunaan	138
7.2.3.1.3	Kesimpulan Pengujian Kebolehgunaan	141
7.2.3.2	Pengujian Ketepatan Laporan	142
7.2.3.2.1	Asas Persekitaran Pengujian Ketepatan Laporan	142
7.2.3.2.2	Aktiviti Pengujian, Keputusan, Pembuktian Dan Kesimpulan	144
7.2.3.3	Pengujian Prestasi	156
7.2.3.3.1	Asas Persekitaran Pengujian Prestasi	157

7.2.3.3.2 Hasil Keputusan Pengujian Prestasi Perisian Plagiarizer 2003	161
7.2.3.3.3 Kesimpulan Pengujian Prestasi	164
7.2.4 Pengujian Pemasangan	165

BAB 8 : PENILAIAN PERISIAN

8.1 Pengenalan	166
8.2 Kekuatan Perisian Plagiarizer 2003	166
8.2.1 Kekuatan Luaran	167
8.2.1.1 Mudah Untuk Digunakan	167
8.2.1.2 Laporan	170
8.2.1.3 Masa Tindakbalas Analisa Yang Munasabah	171
8.2.2 Kekuatan Dalaman	171
8.2.2.1 Algoritma Pengesanan Plagiarisme	171
8.2.2.2 Kebolegunaan semula	173
8.3 Kelemahan Perisian Plagiarizer 2003	173
8.4 Kekangan Perisian Plagiarizer 2003	175
8.5 Cadangan Dan Perancangan Masa Hadapan	177
8.6 Kesimpulan	179
Senarai Rujukan	180
Lampiran/Apendiks	182

SENARAI JADUAL

JADUAL		MUKASURAT
Jadual 1.1	Skedul untuk Pembangunan Perisian Plagiarisme	6
Jadual 2.1	Perbandingan Keseluruhan Perisian Berdasarkan Kajian Joint Information System Commitee	28
Jadual 4.1	Penentuan Tahap Plagiarisme	90
Jadual 4.2	Keperluan Perisian Bagi Membangunkan Perisian Penganalisa Plagiarisme	96
Jadual 4.3	Keperluan Perkakasan Pengguna Untuk Melarikan Perisian Penganalisa Plagiarisme	97
Jadual 6.1	Persekitaran Pembangunan Plagiarizer 2003	115
Jadual 7.1	Masa Yang Diambil Untuk Mahir Dengan Fungsi Asas Plagiarizer 2003	131
Jadual 7.2	Masa Yang Diambil Untuk Mahir Dengan Fungsi Tambahan(<i>Advanced</i>) Plagiarizer 2003	139
Jadual 7.3	Spesifikasi Kes Pengujian Ketepatan Laporan Plagiarizer 2003	143
Jadual 7.4	Jumlah Bilangan Laporan yang Dijana oleh Sistem Perisian Berdasarkan Bilangan Sampel	145
Jadual 7.5	Hasil Keputusan Melalui Pengiraan Manual	146
Jadual 7.6	Hasil Keputusan Kes Pengujian 2 Bagi Aturan A	149
Jadual 7.7	Hasil Keputusan Kes Pengujian 2 Bagi Aturan B	150

SENARAI RAJAH

Jadual 7.8	Hasil Keputusan Kes Pengujian 2 Bagi Aturan C	151
Jadual 7.9	Hasil Keputusan Kes Pengujian 2 Bagi Aturan D	152
Jadual 7.10	Perbandingan Hasil Laporan Yang Dijana Dengan Hasil Pengiraan Manual Untuk Aturan A	154
Jadual 7.11	Spesifikasi Komputer Yang Digunakan Dalam Persekitaran Pengujian	157
Jadual 7.12	Persekitaran Kes Pengujian 1 (Tahap Bebanan Rendah)	158
Jadual 7.13	Persekitaran Kes Pengujian 2 (Tahap Bebanan Sederhana)	159
Jadual 7.14	Persekitaran Kes Pengujian 3 (Tahap Bebanan Tinggi)	160
Jadual 7.15	Hasil Keputusan Kes Pengujian 1 (Tahap Bebanan Ringan)	161
Jadual 7.16	Hasil Keputusan Kes Pengujian 2 (Tahap Bebanan Sederhana)	162
Jadual 7.17	Hasil Keputusan Kes Pengujian 3 (Tahap Bebanan Tinggi)	163

SENARAI RAJAH

RAJAH

MUKASURAT

Rajah 1.1	Carta Gantt Pembangunan Perisian Penganalisa Plagiarisme	7
Rajah 2.1	Contoh Laporan yang Dijana Oleh Perisian CopyCatch	21
Rajah 2.2	Contoh Laporan Berbentuk Jadual Peratusan Plagiarisme Yang Dilaporkan Oleh Perisian CopyCatch	21
Rajah 2.3	Contoh Laporan Berbentuk Peratusan Plagiarisme Yang Dilaporkan Oleh Perisian Wordcheck 2.0	24
Rajah 2.4	Contoh Keseketikaan Perisian EVE2 Yang Sedang Mengesan Plagiarisme Antara Tugas Pelajar Dan Bahan Dari Internet	27
Rajah 2.5	Persekitaran Bahasa C++ Yang Berteraskan Kepada Pengkompil	40
Rajah 2.6	Persekitaran Bahasa Java Yang Berteraskan Kepada Penterjemah	40
Rajah 3.1	Model Air Terjun	46
Rajah 3.2	Model Prototaip	47
Rajah 3.3	Model Air Terjun Dan Prototaip	49
Rajah 4.1	Graf Peratusan Berdasarkan Keutamaan Teknik Plagiarisme	67
Rajah 4.2	Graf Peratusan Responden Berdasarkan Keutamaan Faktor Yang Dipilih	69
Rajah 4.3	Carta Pai Peratusan Responden Berdasarkan Kandungan Dan Jenis Laporan Yang Dipilih	70

Rajah 4.4	Graf Peratusan Responden Berdasarkan Keutamaan Teknik	114
Rajah 4.5	Plagiarisme Yang Dipilih	72
Rajah 4.5	Pecahan Komponen Fungsian Bagi Perisian	77
Rajah 5.1	Carta Alir Fungsi Mengimport Fail Hingga Penjanaan	119
Rajah 5.1	Laporan Plagiarisme	103
Rajah 5.2	Carta Alir Fungsi Bantuan Pengguna Perisian Penganalisa	116
Rajah 5.2	Plagiarisme	104
Rajah 5.3	Rekabentuk Antaramuka Utama Perisian Penganalisa	
	Plagiarisme	106
Rajah 5.4	Rekabentuk Antaramuka Import Fail	108
Rajah 5.5	Rekabentuk Antaramuka Penyelarasan	109
Rajah 5.6	Rekabentuk Antaramuka Keseketikaan Yang Menunjukkan	
	Laporan Peratusan Plagiarisme	110
Rajah 5.7	Rekabentuk Antaramuka Keseketikaan Yang Menunjukkan	
	Laporan Terperinci Kes Plagiarisme Antara 2 Fail Tugas	111
Rajah 5.8	Rekabentuk Antaramuka Bantuan Pengguna	112
Rajah 6.1	Gambaran Interaksi Antara Komponen Kelas Perisian	117
Rajah 6.2	Gambaran Bagaimana Plagiarizer 2003 Membuat	
	Analisa Pengesanan Plagiarisme	118
Rajah 6.3	Pecahan Komponen Fungsian Bagi Plagiarizer 2003	122
Rajah 6.4	Antaramuka Utama Plagiarizer 2003	127
Rajah 6.5	Antaramuka Tetingkap Penyelarasan Plagiarizer 2003	127

Rajah 6.6	Antaramuka Tetingkap Penyelarasan Plagiarizer 2003	128
Rajah 6.7	Antaramuka Laporan Keseluruhan Plagiarizer 2003	128
Rajah 6.8	Antaramuka Laporan Berorientasikan Graf Plagiarizer 2003	129
Rajah 6.9	Antaramuka Laporan Terperinci Plagiarizer 2003	129
Rajah 7.1	Model Pengujian Perisian Plagiarizer 2003	131
Rajah 7.2	Graf Masa Pembelajaran Bagi Kedua-dua Kes Pengujian	140
Rajah 7.3	Graf Bilangan Laporan Dijana Lawan Bilangan Sampel Pengujian	145
Rajah 7.4	Graf Purata Kelajuan Masa Kes Pengujian 1 (Tahap Bebanan Ringan)	161
Rajah 7.5	Graf Purata Kelajuan Masa Kes Pengujian 2 (Tahap Bebanan Sederhana)	162
Rajah 7.6	Graf Purata Kelajuan Masa Kes Pengujian 3 (Tahap Bebanan Tinggi)	163

PENGENALAN

1.2.1. Bahasan Pratik

BAB 1

PENGENALAN

Universiti Malaya

PENGENALAN

1.1 Definisi Projek

Perisian penganalisa plagiarisme ini merupakan satu perisian yang dibangunkan untuk membantu para staf akademik khususnya para pensyarah dan tutor di institusi-institusi pengajian seperti universiti dan kolej untuk mengesan gejala plagiarisme di kalangan tugas para pelajar. Gejala plagiarisme ini perlu diberi perhatian yang lebih supaya tugas para pelajar adalah merupakan hasil daya usaha mereka sendiri dan bebas daripada peniruan di kalangan para pelajar sendiri mahupun sumber yang lain.

Secara konvensional atau kebiasaannya, gejala plagiarisme dikesan secara tidak langsung oleh pensyarah atau tutor yang menilai tugas di mana terdapat dua atau lebih tugas para pelajar yang mempunyai ciri-ciri yang sama dari segi struktur ayat dan perkataan dalam mengemukakan sesuatu isi dan idea. Teknik seperti ini adalah tidak efisien apabila melibatkan kes penilaian tugas pelajar yang sangat banyak. Sebagai contohnya, adalah mustahil untuk mengesan plagiarisme yang berlaku dalam 100 tugas yang dihantar di mana setiap tugas tersebut adalah tidak kurang daripada 30 helai muka surat. Maka dari segi inilah, penggunaan perisian ini dipertimbangkan supaya dapat diaplikasikan sepenuhnya agar gejala plagiarisme dapat dikurangkan.

Perisian yang bakal dibangunkan ini berkebolehan untuk menjana laporan bagi melaporkan analisa yang telah dilakukan oleh sistem perisian ke atas beberapa tugas

pelajar dan turut menunjukkan perenggan dan ayat yang terlibat dengan gejala plagiarisme ini. Maka berdasarkan laporan yang dijana ini para pensyarah atau tutor dapat mengesan dan menentukan samada tugas pelajar tersebut terlibat dengan plagiarisme atau tidak.

Secara tidak langsung, jangkamasa pendek perisian yang bakal dibangunkan ini bertujuan untuk mengenalpasti dan mengesan pelajar-pelajar yang terlibat gejala plagiarisme melalui kesamaan struktur ayat dan perkataan dalam tugas mereka di samping memudahkan proses pengesanan plagiarisme oleh pensyarah atau tutor sementara untuk jangkamasa panjang pula, perisian ini dapat dimanfaatkan supaya dapat mengurangkan bilangan kes plagiarisme dan seterusnya mencegah gejala ini daripada berlaku di kalangan pelajar dalam sesebuah institusi pengajian.

1.2 Motivasi Projek

Perkembangan bidang perisian komputer dan penggunaannya dalam institusi pengajian tinggi menyaksikan lebih ramai pelajar menggunakan komputer dalam menyiapkan tugas yang diberikan oleh pensyarah. Maka, gejala plagiarisme di kalangan para pelajar semakin bertambah mudah dan cepat. Menyedari dan melihat kepada berlakunya gejala plagiarisme di kalangan pelajar dan mahasiswa di institusi pengajian tinggi dalam menyiapkan tugas yang diberikan oleh pensyarah, menimbulkan minat dan motivasi kepada diri ini untuk membangunkan perisian penganalisa plagiarisme ini.

Perisian yang dibangunkan untuk mengesan plagiarisme yang terdapat di dalam pasaran adalah amat sedikit dan juga tidak menepati kehendak pengguna selain melibatkan kos yang tinggi untuk membeli dan menggunakan perisian tersebut. Terdapat juga perkhidmatan dan servis berasaskan web yang ditawarkan untuk mengesan plagiarisme di internet tetapi operasi pengesanan plagiarisme tersebut memakan masa yang lama dan tidak wibawa. Menyedari hakikat ini, saya merasakan perlu untuk membangunkan perisian ini.

Selain daripada itu juga, memandangkan untuk membangunkan perisian ini ia melibatkan kemahiran manipulasi pengaturcaraan dan algoritma yang agak sukar dan berkonsepkan kepada penyelesaian masalah, ia menarik minat dan memotivasikan diri ini supaya dapat menghasilkan perisian yang berkualiti dan mempunyai ciri-ciri dinamik serta efisien berdasarkan apa yang telah dipelajari dan daya kreativiti individu.

1.3 Objektif Projek

Perisian ini dibangunkan untuk kegunaan para staf akademik terutamanya para pensyarah dan tutor di institusi pengajian tinggi. Di antara objektif perisian penganalisa plagiarisme adalah untuk:

- Memudahkan para staf akademik terutamanya para pensyarah dan tutor untuk mengesan gejala plagiarisme yang terdapat dalam tugas di kalangan para pelajar.
- Memendekkan masa pengesanan plagiarisme di dalam tugas para pelajar yang melibatkan kes penilaian tugas yang banyak oleh para pensyarah atau tutor di mana perisian ini berkebolehan untuk mengesan plagiarisme dalam kuantiti amaun tugas yang banyak secara serentak.
- Membekalkan antaramuka pengguna yang mudah digunakan berserta fungsi-fungsi yang berkaitan yang menepati keperluan pengguna.
- Mengelak daripada berlakunya gejala plagiarisme dan peniruan di kalangan pelajar dalam menyiapkan sesuatu tugas serta menyebabkan seseorang individu berfikir dua kali untuk melakukan plagiarisme ke atas tugas yang diberikan.
- Menjadikan para pelajar lebih matang, dinamik dan mempunyai daya usaha sendiri dalam menyiapkan tugas yang diberikan tanpa wujudnya plagiarisme dalam tugas mereka.

1.4 Skop Projek

Skop untuk membangunkan perisian penganalisa plagiarisme boleh dibahagikan kepada dua iaitu :

1.4.1 Skop Pengguna

Membangunkan perisian untuk kegunaan para staf akademik iaitu para pensyarah dan tutor di institusi pengajian tinggi sebagai perisian automasi untuk mengesan plagiarisme dalam tugas para pelajar.

1.4.2 Skop Perisian

- 1) Tugas yang dihantar adalah dalam bentuk dokumen elektronik untuk tujuan pengesanan plagiarisme.
- 2) Membangunkan perisian untuk mengesan plagiarisme dalam tugas pelajar di mana tugas tersebut adalah dalam format teks(ASCII teks) iaitu format pemprosesan perkataan bukan format pengaturcaraan, gambar, bunyi atau lain-lain.
- 3) Membangunkan perisian untuk mengesan plagiarisme dalam tugas pelajar di mana tugas yang hendak dibandingkan mestilah dalam bahasa yang sama.

- 4) Membangunkan perisian di mana bahasa untuk antaramuka pengguna perisian tersebut adalah bahasa Inggeris supaya perisian ini dapat ini digunakan dan diterima-pakai secara meluas oleh pelbagai jenis pengguna.
- 5) Membangunkan perisian yang di spesifikasi penggunaannya untuk mengesan plagiarisme yang melibatkan pengesanan ke atas tugas dalam amaun yang besar dan banyak berbanding tugas tutorial yang sedikit bagi mendapatkan keputusan yang lebih tepat.

1.5 Perancangan Aktiviti Projek

Jadual 1.1 : Skedul Untuk Pembangunan Perisian Penganalisa Plagiarisme

Aktiviti/Fasa	Tarikh Mula	Tarikh Tamat	Jangkamasa
Perancangan	10 Jun 2002	23 Jun 2002	2 minggu
Kajian Literasi	24 Jun 2002	14 Julai 2002	3 minggu
Analisa Keperluan Sistem	15 Julai 2002	11 Ogos 2002	4 minggu
Sessi VIVA	19 Ogos 2002	30 Ogos 2002	2 minggu
Rekabentuk Sistem	12 Ogos 2002	27 Oktober 2002	11 minggu
Pengkodan	28 Oktober 2002	12 Januari 2003	12 minggu
Pengujian & Penilaian	23 Disember 2002	2 Februari 2003	5 minggu
Konsultasi Penyelia	10 Jun 2002	2 Februari 2003	35 minggu
Dokumentasi	18 Jun 2002	2 Februari 2003	34 minggu

Aktiviti/Fasa

Perancangan

Kajian Literasi

Analisa Keperluan

Sessi VIVA

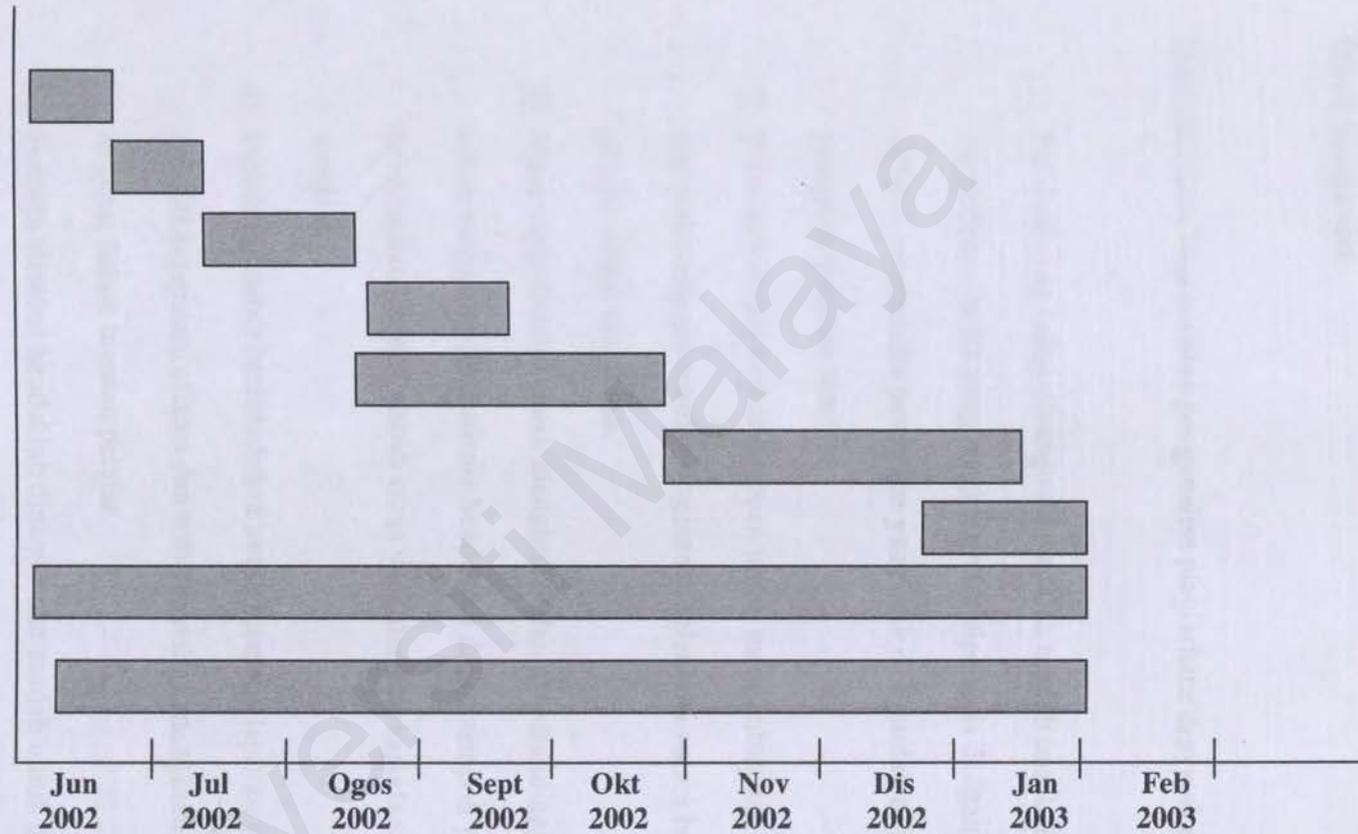
Rekabentuk Sistem

Pengkodan

Pengujian & Penilaian

Konsultasi Penyelia

Dokumentasi



Rajah 1.1: Carta Gantt Pembangunan Perisian Penganalisa Plagiarisme

1.6 Hasil Jangkaan

Hasil jangkaan bagi perisian penganalisa plagiarisme dinyatakan seperti di bawah:

- 1) Perisian yang bakal dibangunkan adalah mudah untuk digunakan dan memerlukan masa yang singkat untuk dipelajari dengan membekalkan ciri-ciri antaramuka pengguna yang baik dan mudah difahami oleh pengguna perisian tersebut.
- 2) Perisian tersebut berkemampuan untuk mengimbas dan membuat analisa dan perbandingan untuk mengesan plagiarisme antara beberapa tugas pelajar dalam satu masa.
- 3) Masa yang diambil untuk mengimbas dan membandingkan tugas pelajar untuk mengesan plagiarisme hendaklah dalam tempoh yang munasabah berdasarkan kepada nisbah masa yang diambil kepada saiz tugas tersebut
- 4) Perisian tersebut berkebolehan untuk menjana laporan yang mempunyai ciri-ciri ketepatan, efisien dan wibawa dalam mengesan plagiarisme yang terdapat dalam tugas pelajar.
- 5) Perisian tersebut hendaklah dinamik dan mudah untuk diperluaskan kebolehmpuan fungsianinya supaya mudah diperbaiki untuk kegunaan masa hadapan.

1.7 Organisasi Bab

Bab 1 : Pengenalan

Bab ini memberikan gambaran secara menyeluruh tentang perisian penganalisa plagiarisme yang akan dibangunkan kelak. Bab ini merupakan fasa awal bagi pembangunan perisian di mana ia memberi penerangan mengenai definisi projek, motivasi projek, objektif projek, skop projek, perancangan aktiviti projek dan hasil jangkaan perisian yang akan dibangunkan.

Bab 2 : Kajian Literasi

Kajian literasi membincangkan perkara-perkara berkaitan dengan gerak kerja awal dalam bab 1. Bab ini memperihalkan mengenai huraian kes masalah dengan lebih mendalam berdasarkan maklumat-maklumat yang didapati, kajian mengenai sistem dan perisian yang sedia ada di pasaran, mengkaji setiap kelebihan dan kekurangan sebelum pembangunan sistem perisian dan juga kajian mengenai bahasa pengaturcaraan yang akan digunakan semasa proses pembangunan.

Bab 3 : Metodologi Pembangunan

Bab ini menerangkan mengenai metodologi yang akan digunakan semasa fasa pembangunan perisian dan justifikasi pemilihan metodologi tersebut. Selain daripada itu juga, ia memperihalkan dengan lebih lanjut mengenai setiap fasa yang terdapat di dalam fasa pembangunan sistem perisian ini.

Bab 4 : Analisa Keperluan

Bab ini memerihalkan mengenai tentang bagaimana maklumat dan fakta di cari bagi mendapatkan keperluan bagi perisian. Selain itu juga, ia turut membincangkan mengenai

hasil tinjauan yang telah dilakukan. Segala maklumat dan hasil tinjauan yang telah dilakukan dirumus dan dianalisis supaya keperluan bagi perisian tersebut didapati.

Analisa keperluan sistem merangkumi segala fungsian dan bukan fungsian merangkumi algoritma pengiraan yang digunakan bagi menentukan peratusan plagiarisme.

Bab 5 : Rekabentuk Sistem

Bab ini menyentuh rekabentuk logikal dan rekabentuk antaramuka bagi perisian yang akan dibangunkan. Rekabentuk logikal merangkumi carta alir yang menunjukkan bagaimana sistem perisian akan digunakan oleh pengguna sementara rekabentuk antaramuka menunjukkan antaramuka prototaip bagi perisian tersebut untuk dinilai oleh pengguna perisian ini.

Bab 6 : Pengkodan dan Perlaksanaan Perisian

Bab ini menyentuh aspek teknikal tentang pendekatan yang dibuat semasa pengaturcaraan dilakukan semasa perisian ini dibangunkan merangkumi pengkodan dan perlaksanaan ke atas perisian.

Bab 7 : Pengujian Perisian

Bab ini menyentuh aspek teknikal pengujian di mana menerangkan pendekatan dan teknik tentang bagaimana pengujian dilakukan ke atas perisian ini bermula daripada pengujian unit sehingga pengujian pemasangan perisian

Bab 8 : Penilaian Perisian

Bab ini menyentuh tentang kekuatan, kelemahan, kekangan yang wujud dalam perisian yang telah dibangunkan serta perancangan dan cadangan untuk masa hadapan.

KAJIAN LITERASI

2.1 Pengantar

Bab ini bertujuan untuk memberi pemahaman awal tentang arti dan definisi literasi yang akan digunakan dalam penelitian yang berkaitan dengan literasi di kalangan masyarakat umum. Selain itu, bab ini juga membahas tentang literasi sebagai salah satu aspek utama yang terlibat di dalam proses-proses politik, sosial, dan budaya lainnya. Adapun definisi literasi yang akan digunakan dalam penelitian ini adalah:

BAB 2

KAJIAN LITERASI

Selain istilah literasi yang sudah dikenal, literasi digital adalah istilah yang lebih baru yang mengacu pada kemampuan yang dibutuhkan untuk menggunakan literasi digital sebagai alat untuk berinteraksi dengan dunia digital dan menggunakan literasi digital untuk meningkatkan kualitas hidup.

Literasi digital adalah kemampuan untuk menggunakan literasi digital sebagai alat untuk berinteraksi dengan dunia digital dan menggunakan literasi digital untuk meningkatkan kualitas hidup. Literasi digital adalah kemampuan untuk menggunakan literasi digital sebagai alat untuk berinteraksi dengan dunia digital dan menggunakan literasi digital untuk meningkatkan kualitas hidup.

Literasi digital adalah kemampuan untuk menggunakan literasi digital sebagai alat untuk berinteraksi dengan dunia digital dan menggunakan literasi digital untuk meningkatkan kualitas hidup. Literasi digital adalah kemampuan untuk menggunakan literasi digital sebagai alat untuk berinteraksi dengan dunia digital dan menggunakan literasi digital untuk meningkatkan kualitas hidup.

KAJIAN LITERASI

2.1 Pengenalan

Bab ini bertujuan untuk memberi penerangan latarbelakang dan kefahaman mengenai bidang atau aspek utama yang terlibat di dalam pembangunan perisian penganalisa plagiarisme ini. Antara skop kajian latarbelakang projek ini merangkumi :

- 1) Kajian ke atas plagiarisme itu sendiri
- 2) Kajian perisian dan aplikasi berasaskan web yang terdapat di pasaran dan analisa ke atas produk tersebut
- 3) Kajian bahasa pengaturcaraan.

Sumber-sumber rujukan bagi membangunkan perisian ini adalah penting bagi menghasilkan perisian yang berkualiti dan menepati kehendak pengguna. Berikut merupakan kaedah dalam penemuan rujukan dan maklumat iaitu :

- Mengadakan sesi perbincangan bersama penasihat projek dan rakan
- Membuat kajian dan penyelidikan daripada internet, jurnal-jurnal elektronik, buku-buku rujukan, majalah dan tesis-tesis pelajar tahun-tahun sebelumnya.
- Analisa ke atas perisian yang terdapat di pasaran melalui internet.
- Membuat pertanyaan melalui e-mel kepada syarikat dan pengaturcara yang membangunkan perisian tersebut

2.2 Plagiarisme

2.2.1 Definisi

Plagiarisme dalam konteks ini merupakan satu gejala negatif yang berlaku di kalangan pelajar-pelajar terutamanya di institusi pengajian tinggi yang semakin bertambah kadar peratusannya dari semasa ke semasa tidak kira dalam mahupun luar negara. Walaupun gejala ini merupakan satu perkara yang tetap tidak boleh dielakkan, ia harus dipandang serius oleh pihak institusi akademik dan langkah pencegahan perlu diambil supaya kadar berlakunya plagiarisme ini dapat dikurangkan.

Plagiarisme boleh didefinisikan sebagai perbuatan mencedok, mengambil dan menggunakan idea, pemikiran, penulisan dan penemuan orang lain menjadi kepunyaan sendiri (Dewan Bahasa dan Pustaka, 1994).

Frasa genting dalam definisi plagiarisme di atas ialah “menjadi kepunyaan sendiri”. Dalam kes pembelajaran di universiti atau mana-mana institusi pengajian tinggi ia adalah merupakan satu perkara yang munasabah dan normal untuk menggunakan pemikiran, idea, penulisan, penemuan orang lain untuk membantu proses pembelajaran dan pemahaman di dalam sesuatu kursus atau perkara. Dalam membuat sesuatu penyelidikan, tugas, projek atau menulis kod program komputer, kita membuat rujukan di internet, buku rujukan, kertas persidangan dan dokumen-dokumen lain untuk

membantu kita menyiapkannya dan ia masih tidak di katakan plagiarisme selagi kita memberi penerangan dan penghargaan atas sumber yang telah digunakan.

Dalam konteks perniagaan atau komersial, undang-undang hakcipta telah dibentuk untuk membantu dan melindungi organisasi atau individu daripada gejala plagiarisme oleh pihak yang tidak bertanggungjawab. Plagiarisme dalam tugas berbentuk teks atau kod aturcara boleh berlaku dalam banyak sektor atau bidang dan tidak tertakluk hanya kepada institusi akademik sahaja. Sebagai contohnya, pada tahun 1988, Barbara Chase-Riboud telah mengemukakan tuduhan plagiarisme kepada mahkamah ke atas pengarah filem Steven Spielberg dan Dreamworks SKG dalam filem yang bertajuk Amistad (Handy, 1997) .Di sini Chase-Riboud mengemukakan kes di mana senario, watak dan aspek yang terdapat dalam buku yang dikarang olehnya iaitu Echo of Lions telah digunakan dan disalin semula secara tidak sah tanpa kebenaran. Walau bagaimanapun pada 9 Februari 1988, Chase Ribaud telah menggugurkan tuduhan kes dan filem tersebut telah ditayangkan.

2.2.2 Mengapa Plagiarisme Berlaku

Dalam kes-kes plagiarisme yang telah berlaku, satu kajian yang spesifik telah dilakukan oleh Joint Information Science Committee (JISC) mengenai gejala plagiarisme berdasarkan prosiding yang telah dibentangkan yang menyatakan antara sebab-sebab berlakunya plagiarisme adalah (Jing & McKeown, 1999) :

- Plagiarisme adalah merupakan satu jalan yang pintas dan mudah yang diambil oleh para pelajar untuk menyiapkan tugas yang telah diberikan.
- Kewujudan teknologi elektronik seperti pemprosesan perkataan dan internet menggalakkan berlakunya gejala plagiarisme ini melalui teknik salin dan tampal iaitu '*copy and paste*'.
- Masa yang singkat serta kekurangan dari segi pengurusan dan perancangan masa menyebabkan pelajar mengambil jalan terakhir untuk terlibat dalam plagiarisme sebagai langkah untuk menyiapkan tugas yang diberikan.
- Tekanan daripada tugas-tugas dari subjek yang lain juga merupakan satu faktor yang menyebabkan pelajar terlibat dengan plagiarisme
- Faktor yang dianggap lemah iaitu tugas yang diberikan kepada pelajar adalah amat sukar menyebabkan pelajar terlibat dengan plagiarisme ini.

2.2.3 Jenis-Jenis Plagiarisme

Secara khususnya, dalam konteks akademik, plagiarisme dapat diklasifikasikan kepada beberapa jenis iaitu :

- 1) Pengambilan dan penyalinan maklumat secara mentah daripada internet
- 2) Pengambilan dan penyalinan maklumat secara mentah daripada bahan rujukan seperti buku teks, kertas persidangan, jurnal-jurnal, majalah dan lain-lain
- 3) Pengambilan dan penyalinan maklumat daripada tugas pelajar lain.

Jenis yang pertama iaitu para pelajar mengambil dan menyalin maklumat secara mentah daripada internet. Cara ini berlaku di mana wujudnya satu koleksi elektronik tugas-tugas pelajar yang ditawarkan oleh beberapa syarikat di mana pelajar hanya perlu menaip dan menyatakan tajuk atau tugas yang diinginkan dan seterusnya laman web tersebut akan mencadangkan beberapa koleksi tugas yang berkaitan dengannya. Pelajar tersebut boleh membeli kertas tugas tersebut dengan harga yang murah secara atas-talian. Gejala seperti ini kurang berlaku terutamanya di Malaysia tetapi banyak berlaku di universiti-universiti luar negara dan gejala ini dipandang serius oleh mereka.

Jenis yang kedua iaitu para pelajar mengambil dan menyalin maklumat secara mentah daripada buku teks, jurnal, majalah, kertas-kertas persidangan yang berkaitan. Kes seperti ini juga turut dianggap sebagai plagiarisme juga kerana pelajar tersebut telah mengambil dan menyalin maklumat secara mentah daripada sumber tanpa mengeluarkan

buah fikiran dan pemahaman atas tugas yang telah diberikan dan seterusnya menjadikan bahan tersebut sebagai hasil kerja sendiri.

Jenis yang ketiga iaitu mengambil dan menyalin maklumat daripada tugas pelajar lain atau dikenali sebagai '*collusion*' (Finlay, 1999). Ini tidak bermaksud seseorang pelajar tidak boleh menolong pelajar yang lain semasa membuat tugas tetapi tugas individu perlu ditulis secara bebas tanpa wujud pergantungan struktur ayat antara teks tugas bagi dua pelajar yang berlainan. Kes seperti ini harus dipandang serius kerana memandangkan tiada usaha yang berpatutan yang dikenakan oleh pelajar tersebut dalam menyiapkan tugas yang telah diberikan selain hanya menyalin secara mentah hasil kerja orang lain.

Maka di sini, skop plagiarisme akan dikecilkan dan difokuskan kepada jenis yang ketiga ini iaitu pengambilan dan penyalinan maklumat daripada tugas pelajar yang lain memandangkan ia merupakan kes atau gejala serius yang perlu dicegah.

2.2.4 Mengenalpasti Plagiarisme dalam Tugas Teks

Terdapat banyak faktor-faktor yang boleh dipertimbangkan untuk digunakan bagi mengenalpasti berlakunya plagiarisme dalam penulisan tugas teks iaitu (Paul, 2000) :

1) Penukaran perbendaharaan kata

Plagiarisme boleh berlaku di mana tugas teks asal diambil keseluruhannya dan kemudian sesetengah perbendaharaan kata yang terdapat di dalam tugas teks asal ditukar bagi mengelakkan kesamaan secara 100% antara tugas tersebut.

2) Ketidakteraturan teks

Jika aliran bagi teks tidak konsisten atau tidak seragam ini menandakan mungkin berlakunya plagiarisme di mana samada penulis tidak menulis menggunakan pemikiran sendiri atau menyalin sebahagian daripada teks tugas yang lain.

3) Tanda baca

Adalah mustahil untuk dua penulis yang berlainan menggunakan tanda bacaan yang sama merangkumi posisi kedudukan atau kaedah penggunaan tanda baca tersebut dalam menulis tugas teks masing-masing.

4) Jumlah kesamaan antara teks

Mungkin berlaku kesamaan dari segi jumlah kesamaan antara teks yang ditulis berdasarkan topik yang sama seperti nama, spesifik domain, terma-terma atau fakta yang mana ini dinamakan **perbendaharaan kata terkongsi**. Tetapi adalah mustahil bagi teks yang ditulis secara bebas mempunyai jumlah kesamaan teks

yang banyak. Satu kajian telah dilakukan oleh pembangun perisian CopyCatch iaitu David Wools, jumlah kesamaan perkataan antara teks yang melebihi 40% adalah normal tetapi jika ia melebihi 70% satu analisis lanjutan perlu dilakukan untuk mengenalpasti samada berlakunya plagiarisme atau tidak (Wools, 1999). Sementara dalam kajian disertasi sarjana yang telah dilakukan oleh Susan Finlay (Finlay, 1999), menyatakan tugas teks yang ditulis secara bebas dalam topik yang sama mungkin mengandungi 50% atau lebih perkataan yang sama tetapi gaya penulisan dan struktur ayat adalah tetap berbeza antara satu penulis dengan penulis yang lain.

5) Kesalahan ejaan

Adalah mustahil jika 2 penulis yang berlainan yang menulis tugas teks secara bebas mempunyai kesalahan ejaan pada perkataan dan posisi yang sama dan jumlah kesalahan ejaan juga sama.

6) Struktur sintaktik bagi teks

Mungkin berlaku plagiarisme jika berlakunya 2 teks yang sama dari struktur sintaktik antara keduanya.

7) Susunan kesamaan antara teks

Jika susunan perkataan dan frasa antara teks adalah sama memungkinkan berlakunya plagiarisme kerana adalah mustahil jika mengemukakan fakta dalam susunan yang sama antara tugas teks bagi 2 penulis yang berlainan

8) Frekuensi perkataan

Mustahil jika 2 teks yang ditulis secara bebas mempunyai frekuensi bilangan perkataan yang sama

2.2.5 Teknik-Teknik Plagiarisme yang Dilakukan

Dalam satu artikel jurnal berdasarkan prosiding yang dibentangkan (Jing & McKeown 1999), teknik-teknik plagiarisme yang biasa dilakukan oleh para pelajar antaranya ialah :

- Pengurangan panjang ayat dalam sesuatu perenggan
- Penggabungan ayat antara perenggan atau dari penulis yang berlainan
- Penukaran struktur atau sintaktik ayat
- Menyatakan semula ayat kepada bentuk lain dengan menukar perbendaharaan kata

2.3 Produk Perisian yang Terdapat Di Pasaran

Sebelum membangunkan perisian ini, perbandingan dan analisa mengenai perisian-perisian yang terdapat di pasaran telah dilakukan. Tujuan perbandingan dan analisa tersebut ialah untuk mengkaji kelebihan perisian tersebut merangkumi prestasi, keefisienan sesuatu produk dalam mengesan plagiarisme, antaramuka laporan yang dijana serta skop serta sasaran bagi sesuatu perisian tersebut serta kelemahan yang terdapat di dalam perisian tersebut. Selain daripada itu, ia sedikit sebanyak membantu dalam memberi idea untuk membangunkan perisian ini.

2.3.1 Produk Perisian yang Dipilih untuk Kajian Kes

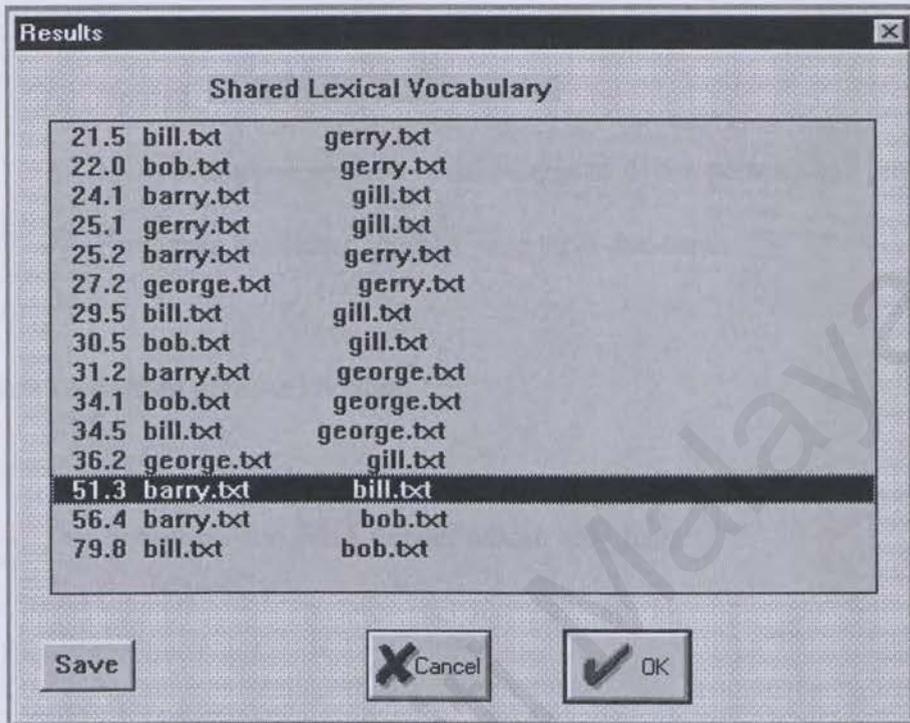
Terdapat 4 produk perisian yang telah dilakukan perbandingan dan analisa iaitu :

1) CopyCatch

CopyCatch adalah produk yang telah dibangunkan oleh David Woolls daripada syarikat CFL Software development (<http://www.CopyCatch.freemove.co.uk>).

CopyCatch merupakan perisian berdiri-sendiri (*'standalone software'*) yang tidak memerlukan capaian web untuk beroperasi. Perisian ini digunakan untuk menentukan samada plagiarisme berlaku antara individu dalam satu set kumpulan tugas yang diuji.. Kesamaan antara tugas teks diukur dari segi kesamaan leksikal antara teks dan

frekuensi perkataan (Wools, 1999). Laporan yang dijana adalah dalam bentuk peratusan kesamaan antara tugas yang telah diuji.



Rajah 2.1 : Contoh Laporan Yang Dijana Oleh Perisian CopyCatch

	Bill and Gerry	George and Gerry	Barry and Bill	Bill and Bob
% Overlap	21.5%	27.2%	51.3%	79.8%
Shared words	103	125	239	418
Shared vocabulary	27	25	69	139
Shared Once	12	8	32	90

Rajah 2.2 : Contoh Laporan Berbentuk Jadual Peratusan Plagiarisme Yang Dilaporkan Oleh Perisian CopyCatch

Kelebihan perisian ini ialah :

- Prestasi pengesanan plagiarisme yang pantas

- Membenarkan pengesanan berlaku dalam satu set tugas yang banyak (proses pengesanan berkelompok)
- Laporan yang jelas dan mudah ditafsirkan dan menyediakan fungsi cetak laporan
- Menyokong penggunaan multi-pengguna dalam persekitaran rangkaian
- Penjana keputusan laporan yang tepat dan cepat

Sementara kelemahan perisian ini ialah:

- Sangat mahal
- Sokongan dari pihak vendor adalah sederhana

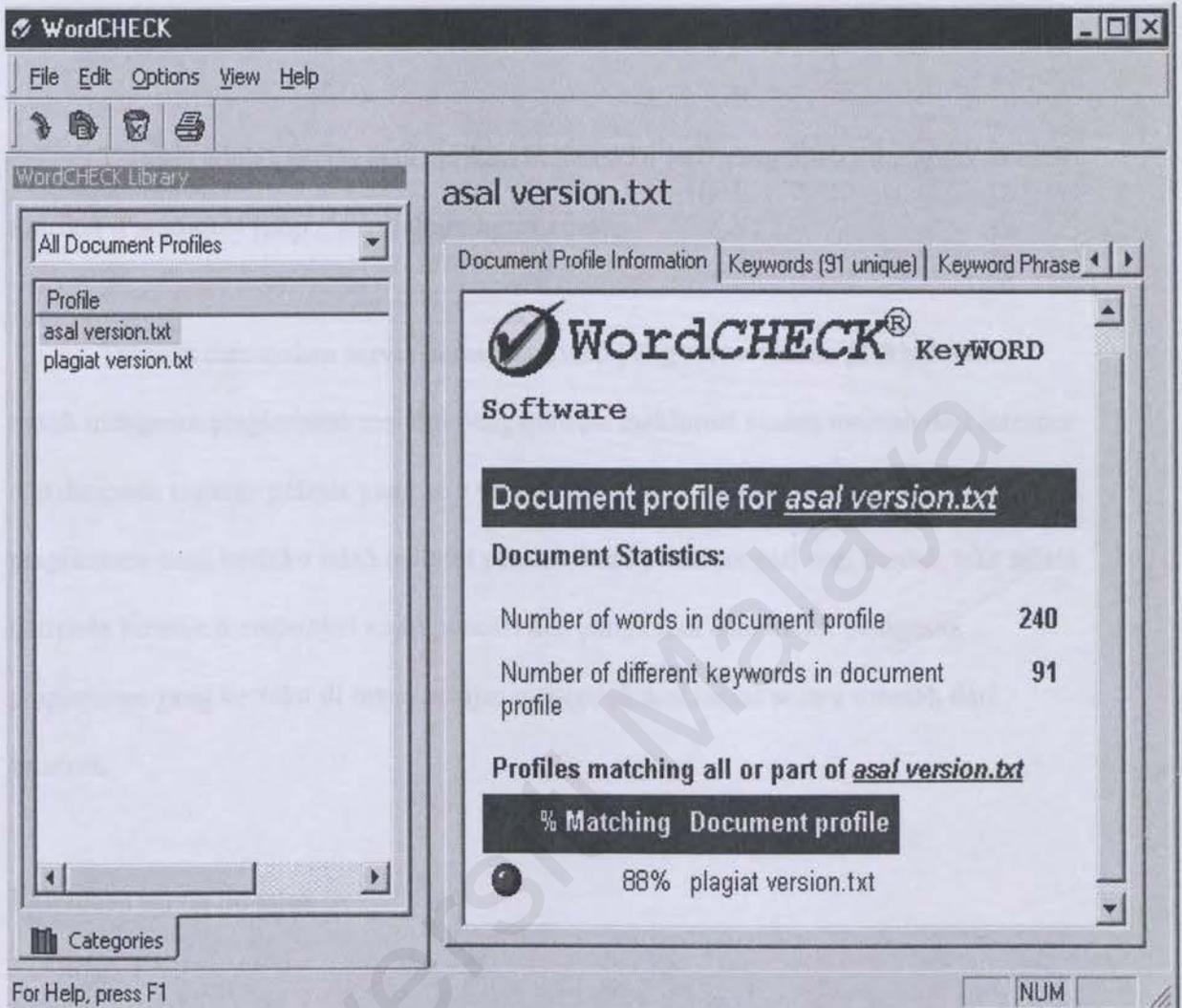
2) WordCheck Keyword 2.0

WordCheck adalah produk yang telah dibangunkan oleh syarikat WordCHECK Systems, Lincoln, Nebraska (<http://www.WordCHECKsystems.com>).

WordCheck Keyword 2.0 direkabentuk dan dibangunkan untuk mengesan plagiarisme antara tugas teks yang diuji dalam satu set kumpulan tugas. Plagiarisme yang berlaku antara tugas dapat dikenalpasti dengan cara mengenalpasti katakunci bagi setiap tugas yang digunakan dan frekuensi katakunci yang digunakan antara tugas. Laporan peratusan plagiarisme akan dijana berdasarkan setiap kata kunci yang telah dikenalpasti.

Kelebihan perisian ini ialah :

- Prestasi pengesanan plagiarisme yang pantas
- Membenarkan pengesanan berlaku dalam satu set tugas yang banyak (proses pengesanan berkelompok)
- Nisbah kos per pengguna yang munasabah
- Laporan yang jelas dan mudah ditafsirkan dan menyediakan fungsi cetak laporan
- Menyokong penggunaan multi-pengguna dalam persekitaran rangkaian
- Penjanaan keputusan laporan yang tepat dan cepat



Rajah 2.3 : Contoh Laporan Berbentuk Peratusan Plagiarisme Yang Dilaporkan Oleh Perisian Wordcheck 2.0

Sementara kelemahan perisian ini ialah:

- Tahap pengesanan plagiarisme yang kurang tepat
- Sokongan dari pihak vendor adalah sederhana

3) Turnitin

Turnitin adalah servis atau aplikasi berasaskan web yang telah dibangunkan oleh syarikat iParadigms (<http://www.iParadigms.com>).

Turnitin merupakan servis berasaskan web yang menawarkan perkhidmatan untuk mengesan plagiarisme melalui pengambilan maklumat secara mentah dari internet dan daripada tugas pelajar yang lain secara serentak. Cara untuk mengesan plagiarisme yang berlaku ialah melalui pendekatan kesamaan dari segi bentuk teks selain daripada turnitin mempunyai enjin pencari dan pangkalan data untuk mengesan plagiarisme yang berlaku di mana pelajar mencedok maklumat secara mentah dari internet.

Kelebihan servis ini ialah :

- Menawarkan perkhidmatan pengesanan secara serentak iaitu plagiarisme yang berlaku apabila pelajar mengambil maklumat daripada internet secara mentah dari internet dan daripada tugas pelajar yang lain.
- Mempunyai kebolehbawaan yang tinggi kerana boleh dicapai dan digunakan di mana-mana kerana ia berteraskan kepada web
- Penjanaan laporan yang mudah difahami dan ditafsirkan
- Sokongan perkhidmatan yang kukuh daripada vendor

Kelemahan servis ini ialah :

- Pengesanan dan penjanaan laporan plagiarisme yang sangat lambat lebih daripada setengah hari
- Kebolehpercayaan yang rendah kerana ia adalah berasaskan web
- Kos penggunaan servis yang tinggi
- Maklumbalas sokongan teknikal yang lambat

4) Essay Verification Engine 2(Eve2)

Eve2 adalah servis yang berasaskan web yang telah dibangunkan oleh syarikat CaNexus (<http://www.CaNexus.com>).

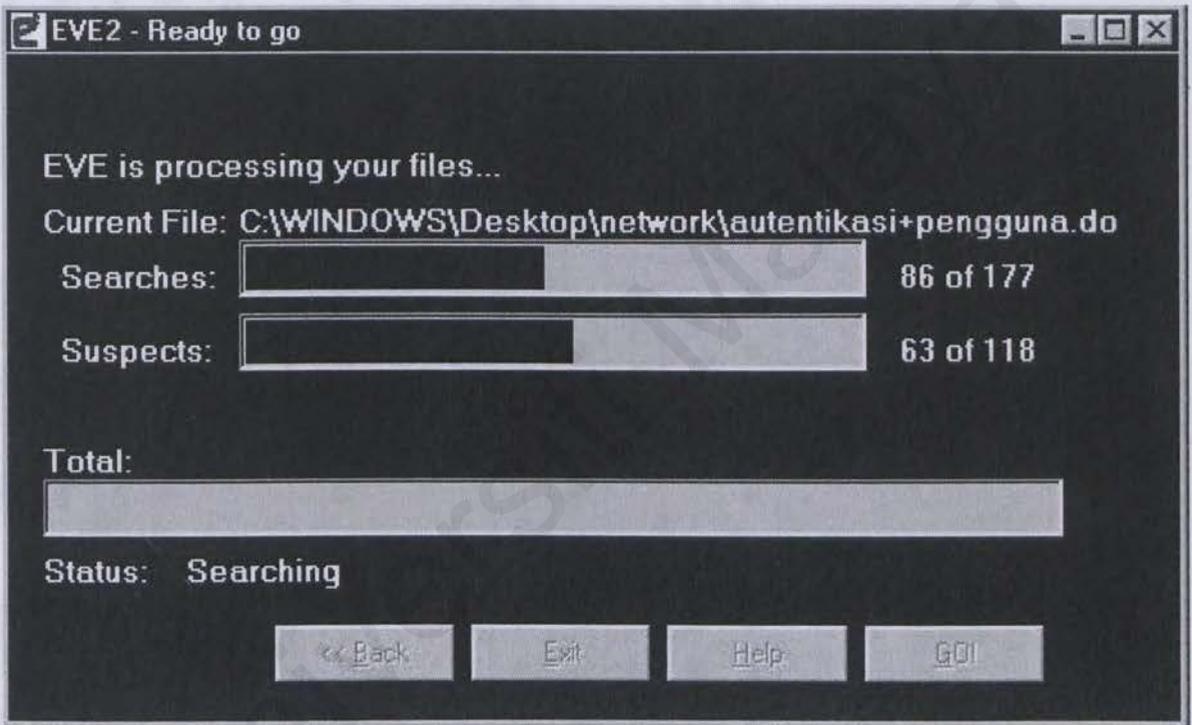
Eve2 dibangunkan untuk mengesan plagiarisme yang berlaku di mana pelajar hanya mengambil maklumat secara mentah daripada internet. Ia tidak menyokong plagiarisme yang berlaku antara tugas pelajar yang mana satu tugas mengambil maklumat daripada tugas pelajar yang lain.

Kelebihan servis ini ialah :

- Masa bagi prestasi pengesanan plagiarisme adalah munasabah dalam masa jangkamasa 20 minit
- Kos penggunaan servis yang munasabah
- Sokongan teknikal daripada vendor adalah kukuh

Kelemahan servis ini ialah;

- Tidak menyediakan sokongan atau servis untuk mengesan plagiarisme antara satu set tugas
- Tidak menyokong penggunaan dalam persekitaran rangkaian
- Kebolehpercayaan yang rendah



Rajah 2.4 : Contoh Keseketikaan Perisian EVE2 Yang Sedang Mengesan Plagiarisme Antara Tugas Pelajar Dan Bahan Dari Internet

Jadual 2.1 : Perbandingan Keseluruhan Perisian Berdasarkan Kajian Joint Information System Committee

	CopyCatch	WordCheck Keyword	Turnitin	Eve2
Fungsi Utama	Plagiarisme antara tugas(<i>collusion</i>)	Plagiarisme antara tugas(<i>collusion</i>)	Plagiarisme antara tugas(<i>collusion</i>)/web	Plagiarisme dari web sahaja
Persekitaran Operasi	Komputer(Windows)	Komputer(Windows)	Aplikasi web	Aplikasi be web
Paras Pengesanan	Frasa dan perkataan	Perkataan sahaja	Frasa dan perkataan	Frasa dan perkataan
Kos	£2000 per institusi £250 per pengguna	US95 per pengguna	US 4000 setahun/tapak	US399.00 per institusi
Pemprosesan Dokumen	Pemprosesan fail tugas secara berkelompok	Pemprosesan fail tugas secara berkelompok	Tidak membenarkan pemprosesan fail secara berkelompok	Pemprosesan fail tugas berkelompok kecil
Prestasi Pengesanan	★ ★ ★ ★ ★	★ ★	★ ★ ★ ★ ★	★ ★ ★ ★
Ketepatan laporan	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★	★ ★ ★
Kebolehpercayaan	★ ★ ★ ★ ★	★ ★ ★ ★	★ ★	★ ★ ★
Mesra Pengguna	★ ★ ★ ★ ★	★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★
Kejelasan Laporan	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★

Sangat Baik ★ ★ ★ ★ ★
 Baik ★ ★ ★ ★
 Sederhana ★ ★ ★
 Memuaskan ★ ★
 Tidak Memuaskan ★

2.3.2 Analisis & Kesimpulan Terhadap Perisian yang Terdapat di Pasaran

Setelah melakukan perbandingan dan analisis ke atas beberapa produk perisian untuk mengesan plagiarisme seperti yang telah dinyatakan dalam bahagian sebelum ini. Kesimpulan atas analisis yang telah dilakukan, apa yang dapat disimpulkan ialah :

- 1) Terdapat hanya sedikit sahaja perisian di pasaran yang mana skop pengesanan plagiarisme adalah antara tugas berbentuk teks yang bertujuan untuk mengesan plagiarisme yang berlaku dalam set tugas pelajar. Secara keseluruhannya apa yang didapati, produk perisian yang kebanyakannya dihasilkan adalah bertujuan untuk mengesan plagiarisme yang berlaku di mana pelajar mengambil maklumat secara mentah di internet. Ini disebabkan kerana permintaan produk perisian sebegini ini adalah tinggi oleh universiti-universiti luar negara memandangkan kes plagiarisme sebegini banyak berlaku di sana.
- 2) Kos merupakan satu faktor yang perlu dipertimbangkan kerana kebanyakan produk perisian yang terdapat di pasaran adalah sangat mahal dan tidak berpatutan berbanding dengan fungsi yang ditawarkan.
- 3) Kebanyakan produk perisian yang terdapat di pasaran adalah lebih berbentuk kepada servis atau perkhidmatan dalam persekitaran aplikasi berasaskan web. Ini adalah amat menyukarkan pengguna kerana ia mengambil masa yang lama untuk mengesan plagiarisme yang berlaku di samping ia mempunyai kebolehpercayaan yang rendah

kerana bergantung kepada prestasi sambungan internet dan ia menjadi satu masalah jika pengguna tidak mempunyai capaian kepada internet.

- 4) Prestasi, ketepatan dan kebolegunaan merupakan faktor major yang menyumbang kepada permintaan pelanggan ke atas sesuatu produk perisian. Maka, prestasi yang cepat, ketepatan yang tinggi serta tahap kebolegunaan yang mudah merupakan kunci utama bagi kebanyakan syarikat dalam membangunkan sesuatu produk.
- 5) Saya telah membuat kajian dan membuat pertanyaan kepada pihak syarikat yang membangunkan perisian yang berkenaan mengenai algoritma yang digunakan. Apa yang didapati melalui jawapan secara kasar oleh pihak syarikat yang membangunkan perisian tersebut ialah :
 - Kebanyakan daripada syarikat menggunakan teknik perbandingan frasa demi frasa berbanding perbandingan perkataan demi perkataan. Ini bertujuan agar keputusan yang didapati adalah lebih tepat.
 - Selain daripada itu, algoritma yang turut digunakan ialah pengiraan frekuensi bilangan katakunci dan perkataan yang digunakan antara satu teks tugas dengan satu teks tugas yang lain untuk tujuan perbandingan
 - Perbandingan antara frasa demi frasa tidak seratus peratus tepat. Oleh itu perbandingan perkataan demi perkataan penting untuk mengesan struktur ayat asal yang ditukar bagi meningkatkan ketepatan.

2.4 Kajian Bahasa Pengaturcaraan

Untuk membangunkan sesuatu perisian maka ia bahasa pengaturcaraan merupakan elemen terpenting supaya perisian tersebut dapat dibangunkan. Untuk membangunkan perisian pengesanan plagiarisme ini, saya telah membuat kajian ke atas 2 bahasa pengaturcaraan yang popular iaitu Java dan C++.

2.4.1 Java

Java merupakan satu bahasa pengaturcaraan yang telah direkabentuk untuk menyelesaikan masalah-masalah domain dalam amalan pengaturcaraan moden. Java ditulis oleh Gosling berdasarkan kepada C++ dengan mengekalkan ciri-ciri minimum yang perlu. Tujuannya agar bahasa yang dihasilkan ringkas memandangkan alat atau mesin untuk melaksanakannya adalah kecil dan mempunyai kuasa pemprosesan yang terhad. Antara kelebihan-kelebihan yang terdapat di dalam Java ialah :

1) Ringkas dan Berorientasikan Objek

Java merupakan salah satu bahasa pengaturcaraan yang mudah dipelajari tanpa perlu menjalani latihan pengaturcaraan yang kompleks dan rumit. Bahasa Java telah direkabentuk sehempir mungkin dengan bahasa C++ supaya pengaturcaraan yang telah biasa dengan C++ boleh bertukar dengan Java dengan lebih mudah dan cepat. Ciri-ciri C++ yang kompleks telah disingkirkan semasa rekabentuk pembangunan bahasa Java ini.

Bahasa pengaturcaraan Java telah direkabentuk supaya berorientasikan objek. Untuk berfungsi di dalam persekitaran rangkaian pelayan-pelanggan yang semakin kompleks dan teragih, bahasa pengaturcaraan harus menggunakan konsep berorientasikan objek. Teknologi Java menyediakan platform pembangunan berorientasikan objek yang kemas dan cekap.

Pengaturcara boleh mencapai pustaka objek tersedia yang terdiri daripada objek yang telah diuji dan menawarkan fungsi-fungsi bermula dari jenis data asas sehinggalah kepada antaramuka rangkaian dan alat bantu antaramuka grafik pengguna. Pustaka ini boleh diperluaskan untuk menyediakan perlakuan baru bagi objek yang telah ditakrif.

2) Selamat dan Teguh

Bahasa pengaturcaraan Java telah direkabentuk untuk mencipta perisian yang mempunyai tahap kebolehpercayaan yang tinggi. Ia menyediakan kemudahan semakan masa kompil yang baik, diikuti dengan semakan paras kedua iaitu semakan masa larian. Model pengurusan ruang ingatannya yang ringkas di mana objek dicipta menggunakan operator 'new'. Penggunaan 'automatic garbage collection' secara tidak langsung menghindarkan ralat pengaturcaraan yang lazimnya dilakukan oleh pengaturcara dalam C/C++ yang boleh menyebabkan kebocoran memori ('leaking memory') yang mana boleh menjejaskan prestasi sistem yang dibangunkan.

Teknologi Java yang mempunyai ciri-ciri keselamatan yang terbina di dalam bahasanya dan masa lariannya membenarkan penulisan aplikasi yang tidak boleh ditembusi dari luar. Di

pentafsiran seperti Java, fasa pautan bagi suatu aturcara menjadi ringkas, bersifat menokok dan kurang bebanan. Oleh itu pengaturcara mendapat manfaat daripada kitar pembangunan yang lebih pantas di mana kaedah pemprototaipan, ujikaji dan pembangunan pantas lebih kerap digunakan berbanding dengan kitar pembangunan konvensional yang terdiri daripada pengkompilan, pautan dan pengujian dengan bebanan yang berat. Dengan mempunyai ciri ditafsirkan, ia juga dapat menyelesaikan masalah pengagihan binari dan versi yang berlainan.

Walaupun prestasi kod bait yang ditafsirkan lazimnya sudah lebih dari memadai, wujud situasi di mana prestasi tinggi diperlukan untuk melaksanakan suatu aturcara. Kodbait diterjemahkan kepada kod mesin unit pemprosesan pusat suatu komputer di mana aplikasi tersebut dilarikan. Namun begitu, sekiranya anda menggunakan pentafsir Java piawai, prestasinya tidaklah boleh dikategorikan sebagai yang terbaik. Oleh itu dengan menggunakan pengkompil tepat-pada-masa ('*just-in-time JIT*') masalah prestasi dapat diatasi. Pengkompil JIT mengubah tugas JVM dengan mengkompil terus kodbait Java kepada kod natif platform yang digunakan. Dengan kaedah ini, prestasi kelajuan pelaksanaan aturcara meningkat kerana kod natif tersebut boleh dilaksanakan terus pada perkakasan.

Keupayaan multi-pembenangan Java menyediakan cara untuk membangunkan aplikasi yang menyokong untuk melarikan beberapa animasi secara serentak dan interaktiviti yang tinggi untuk pengguna. Platform Java menyokong muti-pembenangan pada paras bahasa dengan menambah primitif segerak yang canggih. Pustaka bahasa ini menyediakan kelas '*Thread*' dan sistem masa-lariannya menyediakan pemantau dan primitif penguncian syarat. Tambahan pula pada paras pustaka, pustaka sistem paras-tingginya telah ditulis supaya ia

selamat dari segi pembenangan. Fungsian yang disediakan oleh Java boleh digunakan tanpa berlaku percanggahan walaupun dengan pelaksanaan banyak bebenang serentak.

Walaupun begitu, terdapat beberapa kelemahan dalam bahasa Java ini iaitu :

- 1) Oleh Java kerana menggunakan penterjemah ('*interpreter*'), maka prestasi kelajuan untuk melaksanakan program adalah lambat jika dibandingkan dengan penggunaan bahasa yang menggunakan pengkompil ('*compilers*')
- 1) Kehausan sumber
- 2) Kurang fleksibel berbanding dengan C++
- 3) Kurang kawalan oleh pengaturcara terhadap keseluruhan aturcara program

2.4.2 C++

C++ merupakan bahasa generasi keempat dan merupakan bahasa tahap tinggi selain daripada Java. Bahasa pengaturcaraan C++ merupakan bahasa yang sangat berkuasa, fleksibel dan berprestasi tinggi berbanding dengan Java. Bahasa C++ ini amat bersesuaian untuk membangunkan satu projek perisian yang besar kerana kod aturcara mudah diselenggarakan disebabkan ia adalah berorientasikan objek yang mempunyai kelebihan fungsi-fungsi seperti multi-perwarisan, enkapsulasi, penyaratan operator, polimorfisma dan lain-lain lagi.

C++ di katakan amat berkuasa kerana ia memberi sepenuh kawalan kepada pengaturcara untuk mengawal dan mengaturcara keseluruhan program. Oleh kerana ia berorientasikan objek, kod aturcara mudah untuk diselenggarakan dan ditambah dalam sesuatu projek pembangunan yang besar. C++ adalah berteraskan kepada penggunaan pengkompil ('*compilers*'). Pengkompil ('*compilers*') menukarkan kod program yang ditulis kepada kod mesin. Kod mesin ini kebiasaannya akan dioutputkan ke dalam bentuk yang format fail terlaksana ('*executable file format*') dan apabila program tersebut dilarikan, komputer akan melaksanakan arahan kod mesin secara berjujukan. Oleh kerana kod dalam bentuk kod mesin, maka program tersebut dapat dilarikan dengan pantas dan cepat kerana ia tidak memerlukan penterjemahan semula kepada kod yang ditulis. Maka ini menjamin prestasi yang cepat dan pantas dalam melarikan program yang telah dibangunkan.

C++ direkabentuk berasaskan kepada ciri-ciri bahasa C. Oleh sebab itu, ia mewarisi segala kelebihan dan kekuatan yang terdapat di dalam bahasa C. Penggunaan penuding,

peruntukan pengurusan ruang ingatan, manipulasi bit, 'casting' dalam aturcara menjadikan bahasa C++ ini sempurna dan sesuai kepada pengaturcara yang mempunyai pengalaman dan kemahiran tinggi mengawal segenap aliran data di dalam aturcara tersebut. Maka dengan sebab itu ia amat bersesuaian dalam menyelesaikan masalah-masalah aturcara yang berkaitan dengan bentuk-bentuk kes penyelesaian masalah ('*problem-solving case*') dan juga dalam membangunkan algoritma yang lebih mendalam dan penggunaan objek secara bebas untuk menyelesaikan masalah tersebut.

Selain itu juga, C++ adalah modulariti kerana segmen kod boleh disimpan di dalam perpustakaan yang ditakrifkan supaya boleh diguna-semula untuk kegunaan masa akan datang. Ia juga fleksibel berbanding bahasa Java kerana membenarkan multi-perwarisan berbanding Java yang menggunakan perwarisan tunggal. Selain itu juga C++ membenarkan penyaratan operator untuk meningkatkan ciri-ciri dinamik yang terdapat di dalamnya.

Walaupun begitu, terdapat juga kelemahan-kelemahan yang terdapat di dalam C++ antaranya ialah :

- 1) Bahasa C++ ini merupakan bahasa yang sukar dipelajari dan tidak sesuai digunakan untuk pengaturcara yang baru mengenali dunia pengaturcaraan kerana terlalu banyak kawalan dan perhatian perlu dipertimbangkan semasa menulis skrip aturcara. Sebagai contohnya pengaturcara perlu berhati-hati dalam menguruskan peruntukan ruang ingatan kepada pembolehubah supaya

tidak wujudnya kebocoran memori ('*leaking memory*') yang boleh menjejaskan prestasi sesebuah sistem atau perisian.

2) Selain itu juga penggunaan penuding, manipulasi bit, '*casting*' dan lain-lain fungsi yang kompleks yang digunakan perlu dikawal dengan berhati-hati dan rapi kerana ralat mudah berlaku akibat kelemahan kawalan terhadap kod aturcara selain C++ kekurangan penyemakan ke atas ralat.

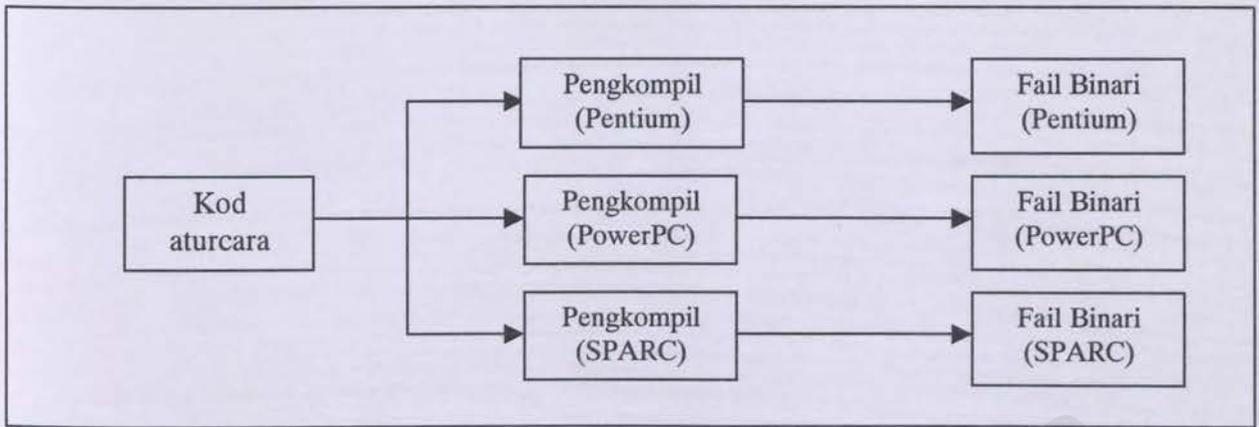
3) C++ juga mempunyai tahap kebolehbawaan yang rendah kerana kod aturcara perlu dikompil semula untuk setiap platform perkakasan yang baru atau berlainan seperti mesin Intel Pentium, SPARC atau PowerPC.

2.4.3 Kesimpulan

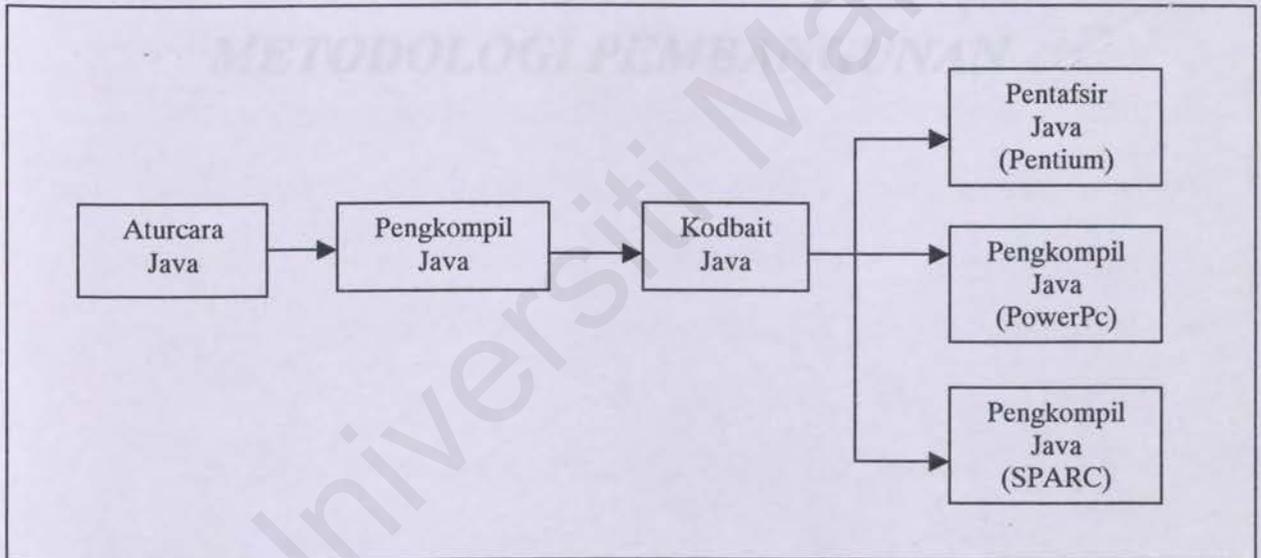
Java merupakan bahasa yang ringkas, mudah dan sesuai untuk dipelajari oleh pengaturcara yang baru. Java menjadikan kerja pengaturcaraan menjadi bertambah mudah di samping ia membekalkan fungsi-fungsi yang berguna untuk mengurangkan ralat serta ia adalah bahasa yang berorientasikan objek. Aplikasi Java adalah berpaksikan kepada penggunaan penterjemah (*'interpreter'*) oleh itu ia amat sesuai dan ideal untuk aplikasi dalam persekitaran rangkaian yang sentiasa memerlukan bahasa yang bebas senibina komputer supaya menjamin tahap kebolehbawaan sesuatu aplikasi perisian yang dibangunkan.

C++ juga merupakan bahasa yang berkuasa dan berorientasikan objek. Ia amat sesuai digunakan untuk membangunkan perisian yang besar dan tugas pembangunan misi kritikal. Selain daripada itu juga, ia amat sesuai dalam menyelesaikan kerja-kerja pengaturcaraan yang melibatkan kes penyelesaian masalah dan juga pembangunan algoritma yang lebih mendalam untuk menyelesaikan masalah tersebut. Oleh kerana ia berteraskan kepada penggunaan pengkompil (*'compilers'*) ia menjamin prestasi yang tinggi ke atas program yang dilarikan.

Rajah 2.4 - Perbandingan Antara Java Yang Berorientasikan Objek Dengan C++



Rajah 2.5 : Persekitaran Bahasa C++ Yang Berteraskan Kepada Pengkompil



Rajah 2.6 : Persekitaran Bahasa Java Yang Berteraskan Kepada Penterjemah

METODOLOGI PEMBANGUNAN

1.1 Pengantar

Metodologi pembangunan merupakan proses penelitian dan juga proses pembangunan yang berorientasi pada pembangunan manusia.

BAB 3

METODOLOGI PEMBANGUNAN

Pemilihan metodologi proses belajar didasarkan dengan sistem atau prosedur yang ingin dikembangkan bagi rumah, di samping itu metodologi pembangunan diarahkan yang utamanya dapat menghasilkan produk yang berorientasi tinggi dan secara tidak langsung tidak mengabaikan aspek-aspek yang lain yang tidak dapat diabaikan.

METODOLOGI PEMBANGUNAN

3.1 Pengenalan

Metodologi pembangunan merangkumi proses pemodelan dan juga proses mengenalpasti setiap aktiviti yang terlibat dalam fasa pembangunan perisian.

Memodelkan proses pembangunan sistem atau perisian merupakan suatu elemen yang terpenting dalam menghasilkan suatu sistem yang sempurna dan berkualiti serta membantu untuk melihat sejauh mana kemajuan pembangunan sistem tersebut telah dicapai. Pemodelan proses bertujuan untuk memberikan pemahaman yang menyeluruh ke atas pembangunan sistem atau perisian merangkumi aktiviti-aktiviti, sumber-sumber dan kekangan yang akan dihadapi semasa pembangunan sistem tersebut. Selain daripada itu juga, ia dapat membantu dalam mengenalpasti ketidakkonsistenan, pertindihan dan pengabaian semasa proses pembangunan. Apabila masalah-masalah tersebut disedari dan diperbetulkan dari awal pembangunan, proses pembangunan menjadi lebih efektif dan boleh menumpukan terhadap pembangunan produk akhir supaya dapat menghasilkan suatu produk yang berkualiti.

Pemilihan pemodelan proses haruslah disesuaikan dengan sistem atau perisian yang ingin dibangunkan bagi mencapai objektif dan matlamat pembangunan sistem yang seterusnya dapat menghasilkan produk yang berkualiti tinggi dan secara tidak langsung ia tidak melebihi kos dan masa yang telah diperuntukkan.

3.2 Metodologi Pembangunan Sistem

Proses ditakrifkan sebagai satu siri langkah-langkah yang melibatkan aktiviti, kekangan dan sumber-sumber yang akan menghasilkan output yang diinginkan. Proses pembangunan perisian juga disebut sebagai kitar hayat perisian (*'software life cycle'*). Metodologi pula merupakan kaedah-kaedah yang digunakan dalam pembangunan sistem bagi mencerminkan matlamat sebenar sistem. Metodologi terdiri daripada proses-proses, yang mana terdiri daripada 7 kriteria iaitu (Pfleeger, 1998) :

- 1) Menerangkan tentang aktiviti-aktiviti proses utama
- 2) Proses menggunakan sumber yang berkaitan dengan kekangan
- 3) Menerangkan tentang subproses yang berkaitan
- 4) Aktiviti untuk proses mempunyai kriteria masuk (input) dan keluar (output)
- 5) Aktiviti dilakukan dalam peringkat atau jujukan
- 6) Mempunyai set prinsip yang menerangkan matlamat setiap aktiviti
- 7) Kekangan atau kawalan boleh dikenakan terhadap aktiviti, sumber atau produk

Metodologi dan analisis ini digunakan untuk mewujudkan pemahaman yang sama dan menyeluruh terhadap aktiviti, sumber dan juga kekangan. Di samping itu, ia juga berfungsi untuk mencari ketakkonsistenan, lewahan, dan pengabaian di dalam

sistem. Dengan ini dapatlah disimpulkan, bahawa metodologi merupakan garis panduan kepada usaha pembangunan suatu sistem. Ia juga dikenali sebagai kitar hayat sistem di mana suatu set kaedah bermula dengan set keperluan pengguna dan menghasilkan sebuah sistem yang memenuhi kesemua keperluan yang dijangkakan.

3.3 Kajian Pemodelan Proses

Untuk membangunkan perisian ini, saya telah membuat kajian ke atas dua pemodelan proses yang popular iaitu model air terjun dan model prototaip.

3.3.1 Model Air Terjun

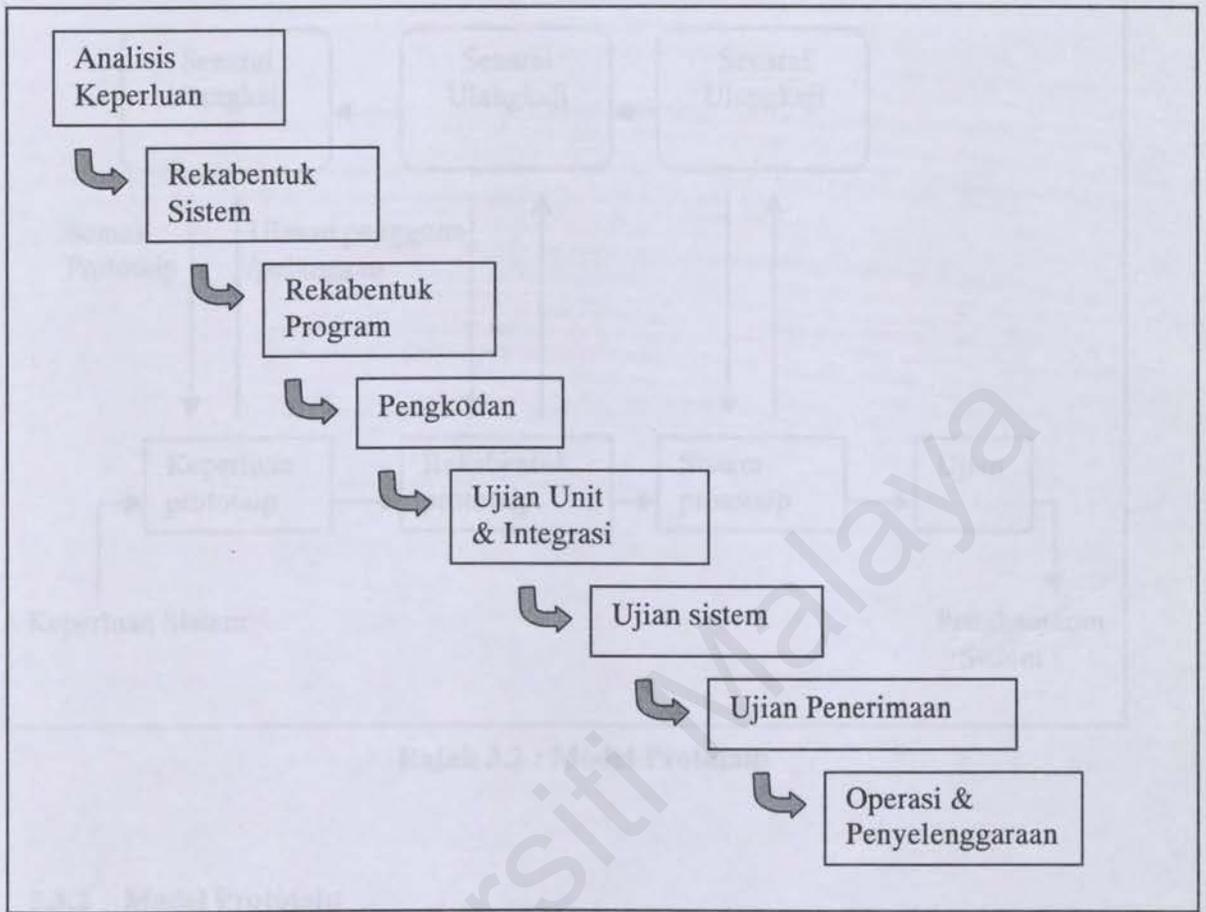
Model ini terdiri daripada beberapa fasa proses pembangunan yang bermula dari satu proses kepada proses seterusnya secara sehala seperti yang digambarkan sebagai air terjun seperti yang ditunjukkan dalam Rajah 3.1. Berdasarkan kepada model ini, dapat digambarkan bahawa satu fasa atau proses di dalam suatu pembangunan perlu disempurnakan terlebih dahulu sebelum langkah seterusnya bermula. Sebagai contohnya, apabila kesemua keperluan telah dikumpul dan didokumentasikan, aktiviti rekabentuk sistem akan dimulakan. Metodologi ini digunakan untuk membangunkan sistem yang tidak kompleks dan tidak melibatkan pengguna sistem yang ramai.

Antara kelebihan-kelebihan bagi model air terjun ini ialah :

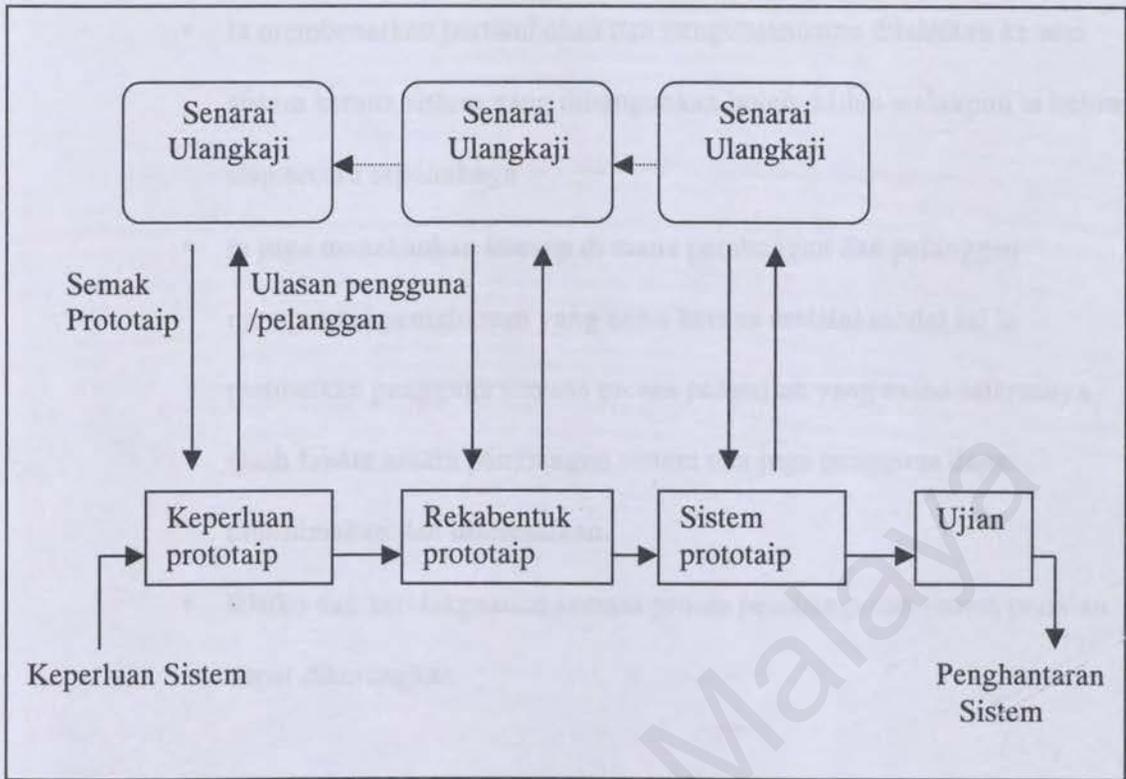
- 1) Model ini menggambarkan fasa-fasa pembangunan suatu sistem atau perisian secara tahap tinggi yang mana membantu memudahkan pembangun sistem atau perisian mengenalpasti aktiviti-aktiviti di dalam fasa pembangunan secara berjujukan dan jelas.
- 2) Secara tidak langsungnya juga, ia merupakan model yang mudah untuk diterangkan kepada pelanggan yang tidak biasa dengan pembangunan sistem atau perisian kerana ia memberi gambaran secara tahap tinggi.
- 3) Pengurus projek boleh menggunakan model ini untuk mengukur jarak masa untuk menyiapkan projek pada satu-satu masa yang diberi di dalam sesuatu fasa.
- 4) Oleh kerana sistem ini berbentuk air terjun, maka sebarang ralat atau kesilapan yang dilakukan pada aktiviti adalah mudah untuk dikesan di dalam satu-satu fasa pembangunan.
- 5) Model air terjun merupakan satu pemodelan proses yang mudah difahami dan mudah untuk digunakan dan kebanyakan para pembangun perisian atau sistem tahu menggunakannya

Walaupun terdapat beberapa kelemahan pada model ini, antaranya termasuklah :

- 1) Tidak memberi gambaran secara jelas tentang bagaimana cara kod dan suatu fasa dibangunkan kecuali jika sistem atau perisian tersebut telah benar-benar difahami.
- 2) Tiada gambaran yang jelas tentang masa setiap aktiviti bertukar dari satu bentuk ke bentuk yang lain. Dengan ini pengurus atau pembangun sistem tidak mempunyai panduan bagaimana untuk menangani masalah perubahan produk atau aktiviti yang mungkin timbul semasa fasa pembangunan.
- 3) Gagal menjadikan perisian sebagai satu proses penyelesaian masalah kerana model air terjun adalah hasil daripada pembangunan perkakasan bukan daripada pembangunan perisian.
- 4) Fasa-fasa yang terdapat dalam model ini tidak melibatkan proses yang berulang kali semasa dalam pembangunan dan proses pengulangan hanya dilakukan apabila terdapat sebarang kesalahan atau ralat bagi sistem yang telah dibangunkan.
- 5) Model air terjun tidak boleh diaplikasikan untuk ke semua jenis pembangunan perisian atau sistem



Rajah 3.1 : Model Air Terjun



Rajah 3.2 : Model Prototaip

3.3.2 Model Prototaip

Model prototaip ini mempunyai beberapa kelebihan dari segi pembangunan di mana ia membenarkan semua atau sebahagian daripada sistem dibina dengan cepat untuk memahami isu-isu yang timbul. Ia berpaksikan kepada konsep membantu memahami masalah sebelum implementasi penyelesaiannya dalam pembangunan sesuatu projek perisian

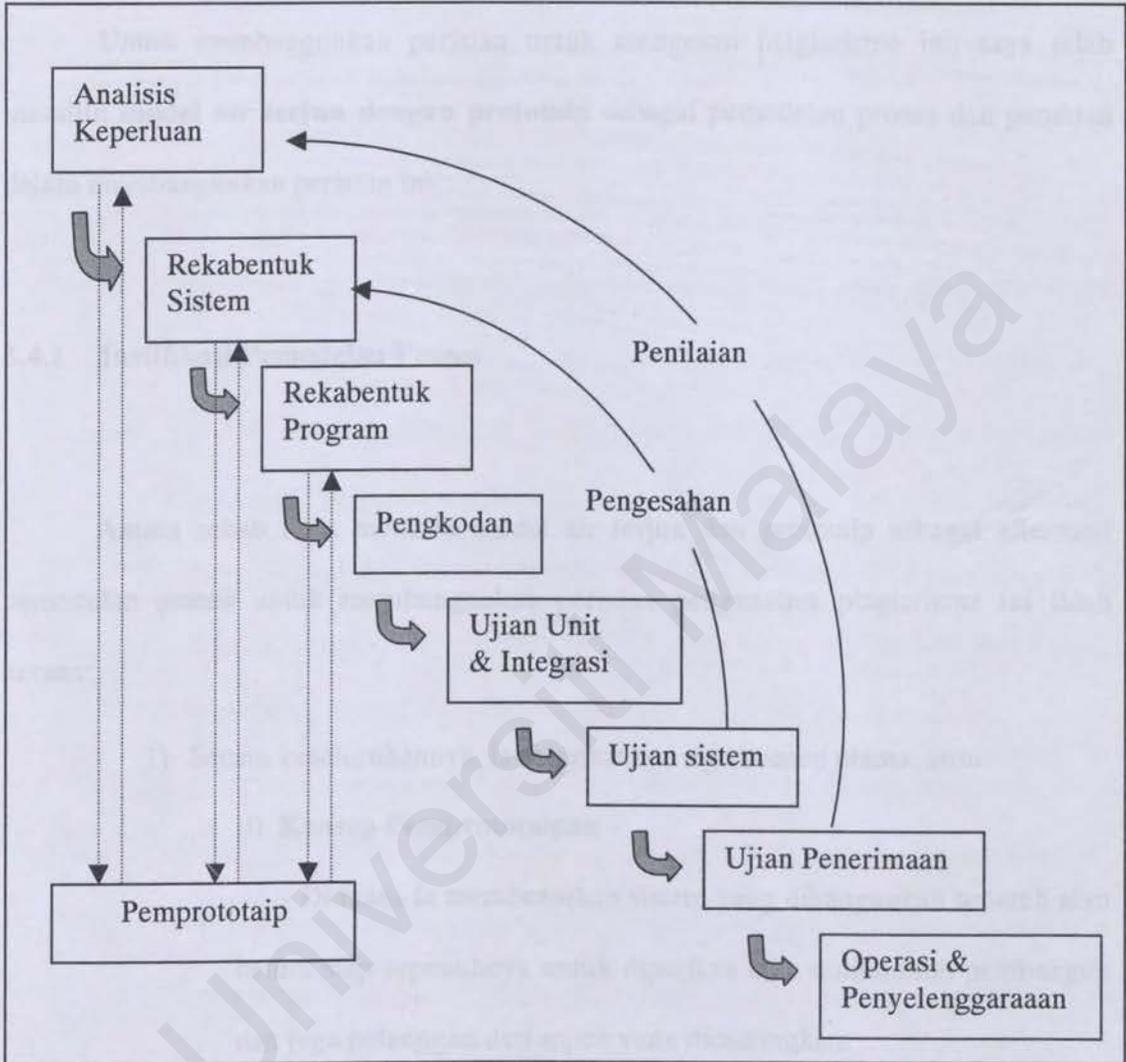
Antara kelebihan model prototaip ialah :

- Ia membenarkan pertambahan dan pengubahsuaian dilakukan ke atas sistem kerana sistem yang dibangunkan boleh dilihat walaupun ia belum siap secara sepenuhnya
- Ia juga menekankan konsep di mana pembangun dan pelanggan mempunyai pemahaman yang sama kerana melalui model ini ia melibatkan pengguna semasa proses pengujian yang mana seterusnya salah faham antara pembangun sistem dan juga pengguna dapat diminimakan dan diselesaikan.
- Risiko dan ketidakpastian semasa proses pembangunan sistem perisian dapat dikurangkan

Walaupun model prototaip ini boleh dikatakan di antara metodologi yang baik, tetapi ia masih mempunyai beberapa kelemahan. Antara kelemahannya ialah :

- Sistem biasanya disiapkan dalam keadaan yang tergesa-gesa oleh itu tiada siapa yang mempertimbangkan kualiti sistem dan penyelenggaraan jangka panjang. Manakala kemungkinan dalam pemilihan sistem pengendalian yang tidak bersesuaian juga mungkin berlaku kerana bagi mendapatkan hasil yang cepat.
- Menggunakan sumber yang banyak kerana ia melibatkan banyak prototaip di mana jika prototaip yang dibangunkan tersebut gagal maka prototaip yang lain perlu dibangunkan sehingga menepati kehendak pengguna dan pelanggan dan ini adalah tidak kos efisien.

3.4 Pengujian dan Prototipe



Rajah 3.3 : Model Air Terjun dengan Prototaip

3.4 Pemilihan Pemodelan Proses

Untuk membangunkan perisian untuk mengesan plagiarisme ini, saya telah memilih **model air terjun dengan prototaip** sebagai pemodelan proses dan panduan dalam membangunkan perisian ini.

3.4.1 Justifikasi Pemodelan Proses

Antara sebab saya memilih model air terjun dan prototaip sebagai alternatif pemodelan proses untuk membangunkan perisian penganalisa plagiarisme ini ialah kerana:

1) Secara keseluruhannya, ia menekankan tiga konsep utama, iaitu:

(i) Konsep Pemrototaipan

- Di mana ia membenarkan sistem yang dibangunkan separuh atau belum siap sepenuhnya untuk diperiksa atau dinilai oleh pembangun dan juga pelanggan dari aspek yang dicadangkan.

(ii) Konsep Penilaian

- Bagi memastikan sistem telah melaksanakan semua keperluan.

(iii) Konsep Pengesahan

- Bagi memastikan setiap fungsi berjalan dengan betul dan tepat.

Berdasarkan kepada 3 konsep ini, ia adalah memadai untuk menghasilkan satu perisian yang berkualiti kerana ia dapat mengurangkan

ketidaktentuan dan salah faham antara pembangun, pengguna dan pelanggan. Penekanan kepada konsep penilaian dan pengesahan amat penting agar segala keperluan bagi sistem tersebut dilaksanakan dengan tepat dan sempurna seperti yang dikehendaki oleh pengguna dan pelanggan.

- 2) Model ini dipilih berdasarkan kelebihan gabungan antara dua 2 model iaitu model air terjun dan juga model prototaip. Oleh itu berdasarkan kepada model air terjun penekanan lebih diberikan kepada pembangunan sistem berfasa, manakala bagi model prototaip pula penekanan lebih tertumpu kepada penglibatan pengguna secara langsung dan tidak langsung serta kebolehan untuk melakukan ubahsuaian dan penambahan semasa sistem dibangunkan yang mana seterusnya pembangunan sistem bertambah mudah dan tidak lari daripada kehendak pengguna.
- 3) Dengan menggunakan kaedah ini, antaramuka pengguna dibina dan diuji sebagai prototaip. Oleh itu, pengguna akan lebih memahami akan ciri-ciri sistem yang dibangunkan dan sebarang ubahan mengikut kehendak pengguna atau pelanggan juga boleh dilakukan bagi memperolehi kepuasan dari pihak pengguna atau pelanggan. Melalui metodologi ini juga pembangun akan lebih prihatin, peka terhadap kesalahan dan ralat serta dapat memahami dengan lebih jelas bagaimana pengguna mahu sistem berinteraksi.
- 4) Di samping itu, pembangun turut mendapat gambaran yang lebih jelas tentang bagaimana sistem ini beroperasi selain daripada penggunaan metodologi ini juga, ia dapat mencetuskan idea baru dan juga

membolehkan penilaian terhadap alternatif strategi rekabentuk serta mengambil keputusan untuk strategi yang paling baik bagi sistem tersebut.

- 5) Dari segi pembangunan sistem berfasa pula, ia akan menghasilkan pengurusan keseluruhan yang lebih sistematik dan dapat memberi gambaran pemindahan proses dan data dari satu fasa ke satu fasa yang lain. Memandangkan metodologi ini menggunakan teknik berfasa maka ia mampu bertindak sebagai dokumentasi atau laporan yang menerangkan apa yang telah dicapai dalam fasa tersebut dan menggariskan satu rancangan untuk fasa yang seterusnya. Penggunaan prototaip pula mampu untuk mengurangkan risiko ketidakpastian kerana sebarang masalah dapat dikesan terlebih dahulu sebelum sistem siap dibangunkan.
- 6) Mungkin ada pihak menyangkal penggunaan model air terjun dan prototaip adalah tidak kos efisien kerana ia akan melibatkan penggunaan sumber yang banyak dalam kes di mana jika wujud beberapa kegagalan dalam prototaip yang dibina akibat ketidakpuasan pengguna. Tetapi harus diingatkan dalam konteks pembangunan perisian penganalisa plagiarisme ini, ia merupakan satu pembangunan perisian yang kecil. Oleh itu ia tidak melibatkan banyak prototaip berbanding dengan pembangunan sistem maklumat. Oleh itu masalah kos dan sumber yang tinggi bukan menjadi persoalan.
- 7) Antara keistimewaan lain menggunakan model ini ialah ia berkeupayaan untuk melaksanakan proses analisis dan model rekabentuk untuk diaplikasikan secara terus dalam proses implementasi. Ini dapat dilihat

kesan daripada penggunaan prototaip pada fasa-fasa tertentu dalam pembangunan perisian penganalisa plagiarisme.

Walaupun begitu, terdapat juga beberapa kelemahan-kelemahan menggunakan pemodelan proses ini antaranya ialah pembangunan sistem perlu dilaksanakan secara berperingkat, di mana setiap fasa perlu dilaksanakan terlebih dahulu sebelum memulakan fasa yang baru. Maka secara tidak langsung sistem tidak boleh dijalankan secara serentak. Rekabentuk proses juga akan selalu berubah memandangkan terdapatnya prototaip dalam fasa-fasa tertentu yang menyebabkan berlakunya pertukaran rekabentuk sekiranya masalah dikesan. Selain daripada itu metodologi ini juga tidak menggambarkan cara kod dibangunkan dalam fasa implementasi yang memungkinkan penyimpangan daripada keperluan sistem yang sewajarnya.

3.5 Fasa –Fasa Pembangunan

Merujuk kepada Rajah 3.3, model air terjun dan prototaip ini terbahagi kepada lapan 8 fasa iaitu:

- Fasa Analisis Keperluan
- Fasa Rekabentuk Sistem
- Fasa Rekabentuk Program
- Fasa Pengkodan
- Fasa Unit Dan Pengujian Integrasi
- Fasa Pengujian Sistem

- Fasa Penerimaan Pengujian
- Fasa Operasi Dan Penyelenggaraan

1) Fasa Analisis Keperluan

Analisis terhadap keperluan dilaksanakan sebagai langkah pertama dalam membangunkan perisian ini. Ini merangkumi penakrifan tujuan, objektif dan juga skop projek sebagai langkah untuk mendapatkan maklumat dan juga gambaran yang lebih jelas mengenai sistem yang hendak dibangunkan kelak. Pada fasa ini juga, keperluan terhadap perkakasan dan juga perisian ditentukan. Antara kaedah yang dilakukan bagi memperoleh maklumat pada fasa ini ialah dengan mengadakan perbincangan bersama penyelia, melayari internet untuk mendapatkan maklumat, tesis-tesis terdahulu, bahan bacaan, soal-selidik ke atas pengguna, temubual samada secara formal atau tidak formal dan juga kajian ke atas perisian yang berada di pasaran serta membuat pertanyaan kepada pembangun perisian tersebut. Berdasarkan kepada data dan maklumat ini, analisa akan dilakukan secara keseluruhan bagi mendapatkan gambaran dan keperluan ke atas perisian yang akan dibangunkan bagi menghasilkan satu perisian yang benar-benar menepati kehendak pengguna dan kompetitif.

2) Rekabentuk Sistem Dan Rekabentuk Program

Rekabentuk logikal terhadap sistem dilaksanakan pada fasa ini. Kaedah ini digunakan sebagai garis panduan kepada pembangunan yang lebih terperinci ke atas perisian ini. Antara rekabentuk yang terlibat ialah :

- Rekabentuk logikal

Rekabentuk yang menggambarkan tentang bagaimana teras dalaman sistem beroperasi

- Rekabentuk antaramuka pengguna

Rekabentuk yang menyatakan tentang bagaimana interaksi antara sistem dan pengguna dijalankan

3) Fasa Pengkodan Perisian

Langkah seterusnya ialah fasa pengkodan yang mana pada fasa ini kod-kod program mula dibangunkan mengikut kajian dan juga garis panduan yang telah diperolehi sebelum ini. Untuk membangunkan perisian ini, teknik yang akan digunakan ialah berkonsepkan kepada pengaturcaraan berorientasikan objek atas tujuan mengoptimumkan kelebihan penggunaan kelas, polimorfisma, perwarisan dan enkapsulasi bagi memudahkan proses pengaturcaraan dan penyelenggaraan.

4) Fasa Unit Dan Pengujian Integrasi

Fasa pengujian dilaksanakan bagi mengesan kesilapan. Matlamat suatu pengujian akan tercapai hanya apabila kegagalan atau kesilapan pada sistem perisian ditemui.

Fasa pengujian unit terdiri daripada beberapa langkah:

- Pemeriksaan kod aturcara
- Pembuktian kod adalah betul
- Pengujian komponen dan sub-komponen program
- Perbandingan teknik

Setiap unit fungsian yang terdapat dalam perisian penganalisa plagiarisme ini akan diuji berdasarkan setiap aspek yang dinyatakan seperti di atas.

Setelah setiap unit fungsian siap diuji, setiap unit fungsian tersebut diintegrasikan dan kemudiannya diuji di mana salah satu pendekatan akan diambil samada menggunakan pendekatan :

- Integrasi atas-bawah (top-down integration)
- Integrasi bawah-atas (bottom-up integration)
- Integrasi big-bang (Big-bang integration)
- Integrasi sandwich (sandwich integration)

Bagi unit-unit yang telah siap, akan diuji kebolehgunaannya pada fasa ini. seterusnya, unit-unit yang telah ditentukan kebolehgunaannya akan diintegrasikan pula dengan unit-unit lain yang telah diuji.

5) Fasa Pengujian Sistem

Pada fasa ini sistem yang dibangunkan akan dipastikan pelaksanaannya yang mana pengujian keseluruhan sistem perisian merangkumi pengujian ke atas fungsi pemasangan, prestasi sistem dan antaramuka pengguna supaya memenuhi spesifikasi yang telah digariskan oleh pengguna atau pelanggan. Pada peringkat ini, pengguna akan terlibat dalam proses pengujian terhadap sistem. Pengguna dalam skop ini merujuk kepada staf akademik iaitu para pensyarah atau tutor. Pengguna boleh memberi komen dan kritikan terhadap sistem yang telah siap. Sekiranya sistem masih tidak menepati apa yang dikehendaki oleh pengguna maka komponen sistem akan diperbaiki, ditambah atau diubah sehingga ia menepati ciri-ciri yang dikehendaki oleh pengguna atau pelanggan.

6) Fasa Ujian Penerimaan

Pada fasa ini ujian dilakukan oleh pengguna untuk memastikan ia telah memenuhi keperluan mereka yang mungkin berbeza dengan kefahaman pembangun. Kadangkala ia dijalankan pada persekitaran sebenar. Sistem yang telah memenuhi spesifikasi yang ditetapkan akan diserahkan kepada pengguna dan pelanggan.

7) Fasa Operasi Dan Penyelenggaraan

Fasa seterusnya ialah fasa operasi dan penyelenggaraan, fasa ini dilaksanakan bagi memantau perjalanan sistem yang beroperasi supaya berjalan dengan lancar. Fasa ini juga dilaksanakan bagi mengesan sebarang ralat atau pepijat yang terdapat pada sistem supaya boleh diperbetulkan dan sistem diperbaiki dari semasa ke semasa.

BAB 4

ANALISA KEPERLUAN SISTEM

Universiti Malaya

ANALISA KEPERLUAN SISTEM

4.1 Pengertian

Sebelum menentukan sebuah program atau aplikasi bagi sebuah sistem atau perantara, kita analisa perlu dilakukan terlebih dahulu bagi mengetahui keperluan sistem atau prosedur tersebut. Apabila analisa keperluan tersebut telah dilakukan, maka kita akan dapat mengetahui dan mengetahui secara terperinci mengenai kebutuhan yang diperlukan kepada sistem tersebut. Keperluan ini meliputi, keperluan pelaksanaan dan juga keperluan sistem logistik.

BAB 4

ANALISA KEPERLUAN SISTEM

4.2 Pengumpulan Fakta

Bagi mendapatkan maklumat mengenai keperluan yang berlaku, beberapa teknik pengumpulan maklumat bagi mendapatkan sumber maklumat mengenai keperluan diperlukan. Antara pengumpulan yang dilakukan ialah:

- 1) Survei
- 2) "Tutuban"
- 3) Layaran intervei

ANALISA KEPERLUAN SISTEM

4.1 Pengenalan

Sebelum menentukan sebarang keperluan atau kelakuan bagi sesuatu sistem atau perisian, satu analisa perlu dilakukan terlebih dahulu bagi mengenalpasti keperluan sistem atau perisian tersebut. Apabila analisa keperluan tersebut telah dilakukan, maka kita akan dapat mengenalpasti dan seterusnya menentukan keperluan bagi sistem tersebut yang merangkumi keperluan fungsian, keperluan bukan fungsian, keperluan perkakasan dan juga keperluan perisian bagi sistem perisian tersebut. Maka, dengan itu saya telah melakukan pengumpulan fakta dan maklumat sebagai satu langkah atau kaedah bagi mengenalpasti dan seterusnya menentukan keperluan untuk perisian ini bagi meningkatkan kualiti perisian yang dihasilkan.

4.2 Pengumpulan Fakta & Maklumat

Bagi menghasilkan satu kajian analisa keperluan yang lengkap, beberapa pendekatan telah dilakukan bagi mendapatkan sumber maklumat mengenai keperluan perisian ini. Antara pendekatan yang diambil ialah :

- 1) Soal Selidik
- 2) Temubual
- 3) Layaran internet

4) **Pertanyaan**

5) **Kajian penyelidikan**

6) **Perbincangan**

4.2.1 Soal Selidik

Kaedah soal selidik yang dilakukan ialah dengan cara mengedarkan borang soal selidik kepada sumber atau pemegang saham yang telah dikenalpasti. Dalam konteks ini, 2 sumber atau pemegang saham telah dikenalpasti yang mana boleh menyumbangkan maklumat kepada pembangunan perisian ini iaitu :

- Pengguna perisian iaitu para pensyarah
- Para pelajar

Dalam konteks skop pengguna perisian, pengguna merupakan orang yang bakal menggunakan perisian yang dibangunkan yang mana di sini fokus diberikan kepada pensyarah-pensyarah yang akan menggunakan perisian ini untuk menganalisa dan seterusnya mengesan plagiarisme yang berlaku dalam set tugas pelajar. Sebanyak 13 borang soal selidik telah diedarkan kepada para pensyarah Fakulti Sains Komputer dan Teknologi Maklumat, Universiti Malaya yang mana ia bertujuan untuk :

- Mengetahui kes plagiarisme yang telah berlaku secara umum serta teknik-teknik plagiarisme yang dilakukan oleh para pelajar dalam menyiapkan tugas yang diberikan

- Mengumpul maklumat mengenai apa yang pengguna perlukan dan tidak diperlukan semasa menggunakan perisian yang bakal dibangunkan.

Selain daripada itu juga, borang soal selidik juga turut diedarkan kepada pelajar-pelajar di mana para pelajar merupakan sumber atau pemegang saham bagi mendapatkan maklumat untuk mengetahui bagaimana mereka melakukan plagiarisme dalam menyiapkan tugas yang diberikan. Berdasarkan maklumat ini, satu kaedah atau teknik yang bersesuaian akan dibangunkan supaya kes plagiarisme yang berlaku dapat dianalisa dan dikesan dengan lebih tepat oleh perisian yang akan dibangunkan. Sebanyak 25 borang soal selidik diedarkan kepada pelajar-pelajar Universiti Malaya atas tujuan tersebut.

Kedua-dua borang soal selidik ini boleh didapati di bahagian apendiks. Hasil analisa ke atas borang soal selidik akan dijelaskan pada seksyen seterusnya iaitu seksyen hasil tinjauan.

4.2.2 Temubual

Temubual merupakan satu teknik untuk mengumpul fakta dan maklumat untuk mendapatkan maklumat yang lebih khusus dan spesifik. Dalam konteks ini, saya telah menjalankan temubual kepada 3 orang pelajar Universiti Malaya sebagai responden bagi mengetahui dengan lebih lanjut dan lebih jelas tentang bagaimana mereka melakukan plagiarisme dalam tugas yang diberikan berbanding dengan kaedah borang soal selidik. Berdasarkan maklumat ini, satu kaedah atau teknik yang

bersesuaian akan dibangunkan supaya kes plagiarisme yang berlaku dapat dianalisa dan dikesan dengan lebih tepat oleh perisian yang akan dibangunkan. Hasil analisa ke atas temubual akan dijelaskan pada seksyen seterusnya iaitu seksyen hasil tinjauan.

4.2.3 Layaran Internet

Internet merupakan gerbang maklumat yang mana kita boleh mendapatkan maklumat global dengan mudah dan pantas. Di sini saya telah melayari internet untuk mendapatkan maklumat mengenai :

- Produk perisian atau perkhidmatan berasaskan web untuk mengesan plagiarisme atau seumpamanya yang terdapat di pasaran global untuk mendapatkan idea dan gambaran ringkas mengenai keperluan umum perisian
- Laporan kajian dan jurnal-jurnal elektronik atas-talian yang dilakukan secara teoritikal dan teknikal terutamanya oleh Joint Information System Committee (JISC) mengenai kes plagiarisme ini.
- Tesis sarjana muda dan disertasi sarjana yang telah dilakukan oleh pelajar-pelajar universiti luar negara untuk mendapatkan idea dan algoritma yang terbaik untuk membangunkan perisian.
- Algoritma fungsian dalaman perisian seperti suffix-trees, algoritma boyer-moore dan lain-lain lagi.

4.2.4 Pertanyaan

Saya juga menggunakan pendekatan melalui pertanyaan sebagai satu teknik untuk mengumpul fakta dan maklumat untuk kegunaan pembangunan perisian. Saya telah membuat pertanyaan melalui e-mel kepada pembangun perisian yang terdapat di pasaran. Pertanyaan ini telah saya ajukan kepada :

- 1) Pembangun perisian CopyCatch iaitu En. David Wools
- 2) Meja pertanyaan turnitin dan WordCheck.

Pertanyaan yang dibuat adalah berkisarkan kepada :

- Apakah ciri-ciri keperluan perisian yang mereka utamakan dalam pembangunan produk perisian mereka
- Algoritma secara umum tentang bagaimana mereka mengesan plagiarisme dalam set tugas para pelajar untuk mendapat keputusan yang lebih tepat.

4.2.5 Kajian Penyelidikan

Penyelidikan yang telah saya lakukan merangkumi penelitian ke atas buku-buku rujukan, jurnal-jurnal dan juga majalah-majalah yang membantu saya mendapatkan maklumat mengenai isu-isu plagiarisme dan juga panduan tentang bagaimana untuk membangunkan perisian.

4.2.6 Perbincangan

Memandangkan tugas untuk membangunkan perisian ini bukan merupakan satu perkara yang mudah maka perbincangan yang kerap bersama penyelia amat penting supaya proses pembangunan perisian dapat dijalankan dengan lancar dan sempurna.

Perbincangan dengan penyelia merangkumi :

- Meminta pandangan, nasihat dan idea untuk pembangunan perisian
- Menyemak semula draf atau prosedur-prosedur bagi setiap bab yang telah dibuat

Perbincangan juga dilakukan bersama rakan apabila menghadapi sesuatu masalah untuk memupuk pemahaman yang lebih mendalam bagi memperbaiki perisian yang akan dibangunkan.

4.3 Hasil Tinjauan

Berikut merupakan hasil bagi setiap tinjauan yang dibuat merangkumi soal selidik kepada para pensyarah dan pelajar serta temubual yang telah dilakukan.

4.3.1 Soal Selidik Pengguna (Para Pensyarah)

Sebanyak 13 borang soal selidik telah diedarkan kepada para pensyarah yang mana borang soal selidik ini terdiri daripada 3 seksyen iaitu :

- Seksyen umum
- Seksyen tugas para pelajar
- Seksyen jangkaan fungsian perisian

Salinan borang soal selidik boleh didapati di bahagian appendiks. Hasil tinjauan daripada borang soal selidik yang diedarkan dipecahkan kepada seksyen-seksyen seperti yang di bawah iaitu:

4.3.1.1 Seksyen Umum

Merujuk kepada seksyen umum dalam borang soal selidik, majoriti pensyarah mengatakan **26-50%** pelajar terlibat dengan kes plagiarisme dalam menyiapkan tugas yang diberikan dan gejala plagiarisme ini semakin bertambah buruk jika tiada langkah

pengecahan dibuat dari sekarang. Kebanyakan responden menyatakan kes plagiarisme yang berlaku adalah dengan cara di mana seorang pelajar **menyalin tugas pelajar yang lain** di mana ini merupakan satu cara plagiarisme yang sangat kritikal berbanding dengan menyalin maklumat daripada internet dan gejala faktor utama berlakunya plagiarisme adalah disebabkan oleh para pelajar yang tidak menguruskan masa dan kerja dengan baik.

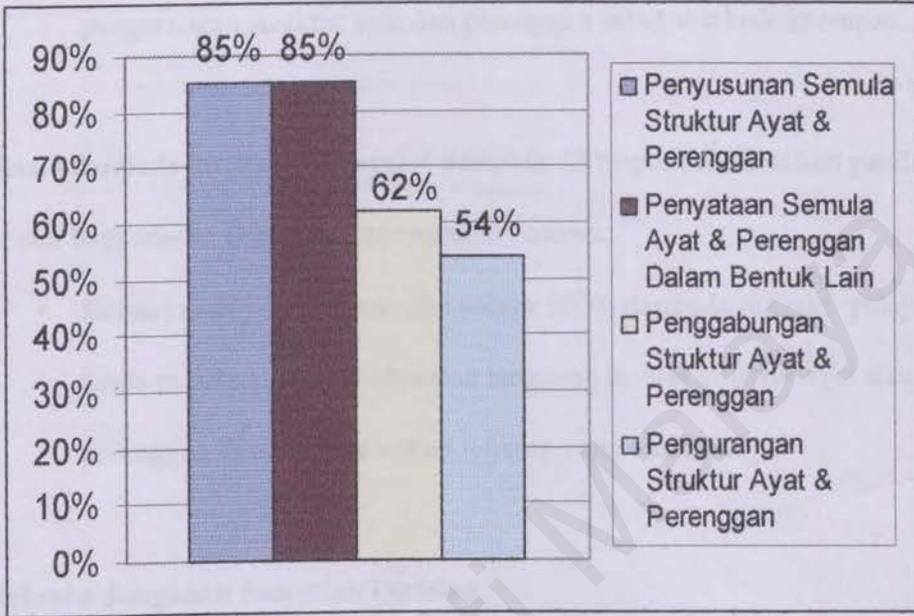
Kebanyakan jabatan dan para pensyarah menyemak ke semua tugas yang dihantar samada terlibat atau tidak dengan gejala plagiarisme tetapi mekanisme penyemakan yang dilakukan adalah secara **manual iaitu secara tidak langsung** menggunakan mata dan tangan. Satu sistem penyemakan berautomasikan komputer tidak digunakan kerana tiada sesiapa yang mula untuk mengusahakannya secara komersial untuk menyemak set tugas yang dihantar dan ramai responden mengatakan satu sistem atau perisian berautomasikan komputer perlu diwujudkan untuk menyemak set tugas para pelajar ini.

4.3.1.2 Seksyen Tugas Para Pelajar

1) Berdasarkan **soalan (9)** dalam borang soal selidik, jenis format fail tugas yang perlu dihantar selain daripada format pemprosesan perkataan (*.doc) ialah :

- Format adobe acrobat (*.pdf)

2) Berdasarkan **soalan (10)** dalam borang soal selidik, teknik plagiarisme yang dilakukan oleh para pelajar untuk menyiapkan tugas yang diberikan mengikut keutamaan susunan ialah :



Rajah 4.1 : Graf Peratusan Responden Berdasarkan Keutamaan Teknik Plagiarisme

Merujuk kepada rajah 4.1 graf menunjukkan :

- Sebanyak 85% atau 11 daripada 13 responden meletakkan teknik menyusun semula struktur ayat dan perenggan sebagai teknik plagiarisme yang utama.
- Sebanyak 85% atau 11 daripada 13 responden meletakkan teknik penyataan semula ayat atau perenggan dalam bentuk yang lain sebagai teknik kedua

- Sebanyak 62 % atau 8 daripada 13 responden meletakkan teknik penggabungan struktur ayat dan perenggan sebagai teknik ketiga
- Sebanyak 54% atau 7 daripada 13 responden meletakkan teknik pengurangan struktur ayat dan perenggan sebagai teknik keempat .

Selain daripada itu juga, sebanyak 4 daripada 13 responden memberi pandangan dan komentar bagi **soalan (10)** yang menyatakan bahawa:

- Kebanyakan pelajar menyalin secara 100% daripada tugas pelajar lain tanpa membuat pengubahsuaian langsung ke atas struktur ayat atau perenggan dalam menyiapkan tugas yang berikan

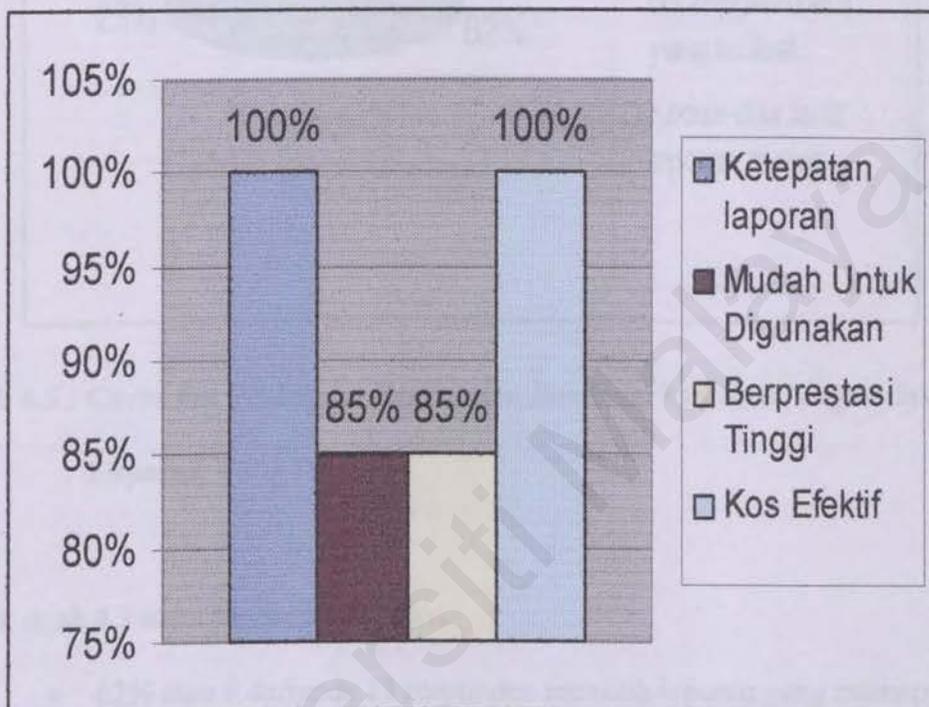
4.3.1.3 Seksyen Jangkaan Fungsian Perisian

1) Berdasarkan **soalan (15)** dalam borang soal selidik, faktor major yang diutamakan oleh pengguna semasa menggunakan perisian untuk mengesan plagiarisme ialah :

Merujuk rajah 4.2 graf menunjukkan :

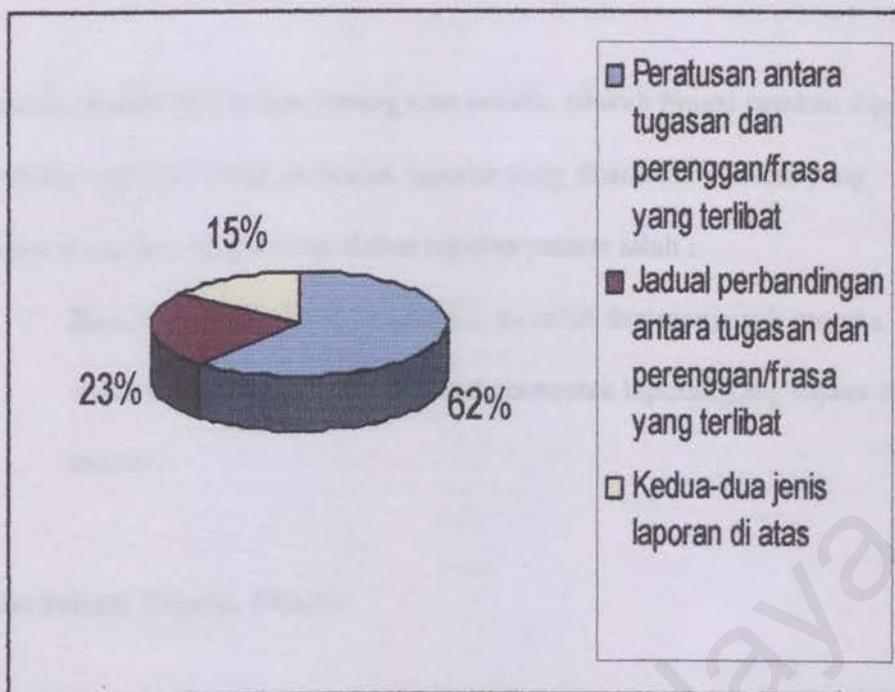
- Kesemua atau 100 %responden meletakkan faktor ketepatan laporan sebagai faktor utama yang diambil kira semasa menggunakan sistem perisian.
- 85 % responden meletakkan faktor mudah untuk digunakan iaitu mesra pengguna sebagai faktor kedua.

- 85 % responden meletakkan berprestasi tinggi iaitu masa tindakbalas yang cepat sebagai faktor ketiga.
- Kesemua atau 100% responden meletakkan kos efektif sebagai faktor keempat.



Rajah 4.2 : Graf Peratusan Responden Berdasarkan Keutamaan Faktor Yang Dipilih

2) Berdasarkan **soalan (16)** dalam borang soal selidik, kandungan atau jenis laporan yang dikehendaki oleh pengguna untuk memaparkan maklumat statistik jika berlakunya plagiarisme dalam set tugas pelajar ialah :



Rajah 4.3 : Carta Pai Peraturan Responden Berdasarkan Kandungan dan Jenis Laporan Yang Dipilih

Merujuk rajah 4.3 carta pai menunjukkan :

- 62% atau 8 daripada 13 responden memilih laporan yang memaparkan peraturan plagiarisme yang berlaku antara set tugas serta perenggan/frasa yang terlibat
- 23% atau 3 daripada 13 responden memilih laporan yang memaparkan jadual perbandingan berdasarkan frekuensi perkataan yang sama dan perenggan/frasa yang terlibat
- 15% atau 2 daripada 13 responden memilih laporan yang mampu untuk memaparkan kedua-dua jenis laporan yang dinyatakan di atas.

3) Berdasarkan **soalan (17)** dalam borang soal selidik, adakah fungsi cetakan diperlukan atau tidak dalam perisian untuk mencetak laporan yang dijana oleh sistem yang menunjukkan wujudnya plagiarisme dalam tugas pelajar ialah :

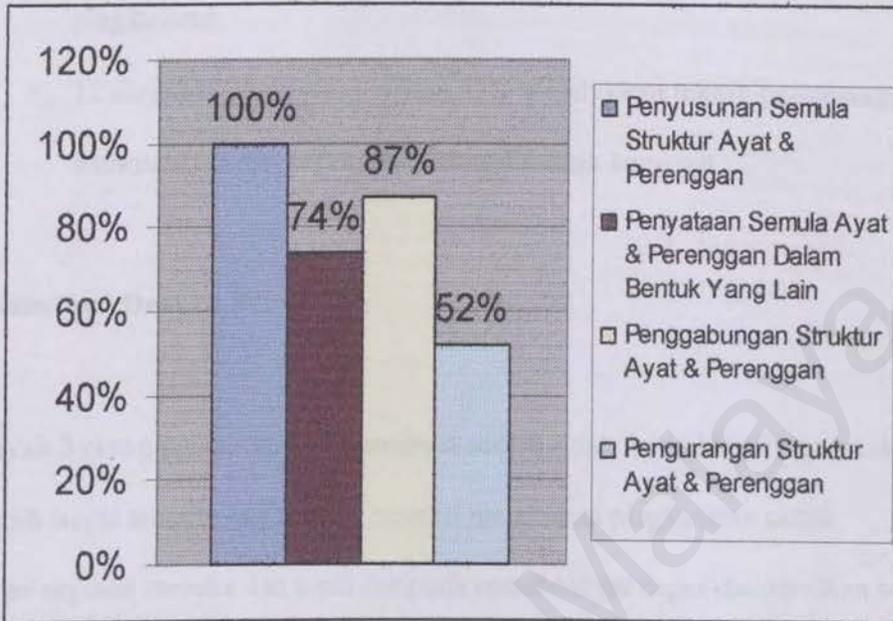
- **Kesemua atau 100 %** responden memilih dan menjawab mereka memerlukan fungsi cetakan untuk mencetak laporan yang dijana oleh sistem

4.3.2 Soal Selidik Kepada Pelajar

Sebanyak 25 borang soal selidik telah diedarkan kepada 25 orang pelajar Universiti Malaya bagi mengetahui bagaimana para pelajar melakukan plagiarisme dalam menyiapkan tugas mereka. Hasil tinjauan adalah seperti berikut :

Daripada 25 responden, 23 daripada mereka menyatakan mereka terlibat dengan gejala plagiarisme. Berdasarkan jawapan responden, sebab utama mengapa mereka terlibat dengan plagiarisme adalah disebabkan oleh tugas yang diberikan oleh pensyarah adalah sukar dan tidak dapat menguruskan masa berdasarkan kesesuaian kerja dengan cekap dan sempurna. Kebanyakan dari responden menyatakan, plagiarisme yang dilakukan dengan cara menyalin daripada tugas pelajar lain iaitu rakan mereka dan menyalin daripada internet. Majoriti responden mengatakan daripada 10 tugas yang diberikan 7-8 tugas biasanya disiapkan dengan melakukan plagiarisme.

Berdasarkan **soalan (6)** borang selidik pelajar, teknik utama atau yang biasa dilakukan oleh responden untuk menyiapkan tugas yang diberikan ialah :



Rajah 4.4 : Graf Peratusan Responden Berdasarkan Keutamaan Teknik Plagiarisme Yang Dipilih

Merujuk rajah 4.4 graf menunjukkan :

- Kesemua responden atau 100 % meletakkan teknik penyusunan semula struktur ayat dan perenggan sebagai teknik utama semasa melakukan plagiarisme
- 17 daripada 23 responden atau 74% meletakkan teknik penyataan semula ayat dan perenggan sebagai teknik kedua semasa melakukan plagiarisme

- 20 daripada 23 responden atau 87% meletakkan teknik penggabungan struktur ayat dan perenggan sebagai teknik ketiga semasa melakukan plagiarisme
- 12 daripada 23 responden atau 52% meletakkan teknik pengurangan struktur ayat dan perenggan sebagai teknik keempat

4.3.3 Temubual Dengan Pelajar

Sebanyak 3 orang pelajar telah ditemubual secara tidak formal bagi mengetahui dengan lebih lanjut tentang bagaimana mereka melakukan plagiarisme untuk menyiapkan tugas mereka dan hasil daripada temubual ini dapat disimpulkan seperti di bawah :

- Teknik plagiarisme yang biasa dilakukan oleh responden ialah dengan menyalin keseluruhan tugas pelajar dan kemudian mengubah struktur ayat serta perenggan dengan cara meletakkannya pada perenggan yang lain atau pada mukasurat yang lain dalam satu fail tugas dan kaedah ini merupakan kaedah yang paling cepat dalam menyiapkan tugas yang diberikan
- Mengubah sesetengah perbendaharaan kata supaya tidak kelihatan 100 % sama dengan tugas asal.
- Membuat pengurangan dan penggabungan ayat atau perenggan
- Mengubah paparan visual bagi fail tugas seperti mengubah saiz atau jenis tulisan dan ini juga merupakan cara yang cepat supaya tidak dapat ditangkap melakukan plagiarisme dalam menyiapkan tugas.

4.4 Rumusan Daripada Hasil Tinjauan

Berdasarkan hasil tinjauan yang telah didapati yang merangkumi soal selidik, temubual kepada pensyarah dan pelajar dan kajian literasi yang merangkumi analisa perbandingan ke atas perisian yang terdapat di pasaran, apa yang dapat dirumuskan ialah:

- 1) Suatu sistem perisian yang berautomasikan komputer untuk mengesan plagiarisme yang berlaku dalam tugas pelajar amat diperlukan yang mana faktor yang dipertimbangkan untuk menggunakan perisian tersebut mengikut keutamaan ialah :
 - Ketepatan laporan
 - Mudah untuk digunakan(mesra pengguna)
 - Berprestasi tinggi dengan masa tindakbalas yang minima
 - Kos efektif

- 2) Sistem perisian yang dibangunkan berkemampuan untuk mengesan plagiarisme yang dilakukan secara efisien dan tepat terutama dalam melibatkan teknik-teknik yang di bawah iaitu :
 - Penyusunan semula struktur ayat dan perenggan dengan cara mengubah kedudukan ayat atau perenggan tersebut

- Salinan secara 100% daripada tugas asal berbanding penukaran struktur ayat dan perenggan kepada bentuk yang lain
- Penggabungan dan pengurangan ayat atau perenggan daripada 2 atau lebih tugas asal

Di mana ini menunjukkan bahawa sistem perlu menganalisa keseluruhan fail tugas berbanding analisa secara bahagian demi bahagian bagi menentukan samada berlaku atau tidak plagiarisme dalam set tugas tersebut.

3) Sistem perisian tersebut perlu menjana laporan yang :

- Memaparkan peratusan plagiarisme antara set tugas dan turut memaparkan laporan terperinci mengenai frasa dan perenggan yang terlibat.

4) Jenis format fail tugas yang perlu dianalisa ialah berjenis :

- Pemprosesan perkataan '*Word Documents*' (*.doc)

Dan lebih baik jika sistem perisian berkemampuan untuk menganalisa format fail tugas berjenis:

- Pemprosesan perkataan '*adobe acrobat*' (*.pdf).

5) Sistem perisian berkebolehan untuk mencetak laporan yang telah dijana.

4.5 Keperluan Sistem

Keperluan adalah suatu ciri sistem atau penerangan tentang sesuatu yang boleh dilakukan oleh sistem bagi memenuhi tujuan sistem tersebut. Biasanya keperluan dapat dibahagikan kepada tiga kategori iaitu :

- 1) Keperluan yang perlu dipenuhi dan dicapai secara mutlak
- 2) Keperluan yang boleh diadakan tetapi tidak diperlukan
- 3) Keperluan yang mungkin diperlukan tetapi boleh ditiadakan jika tidak berkenaan.

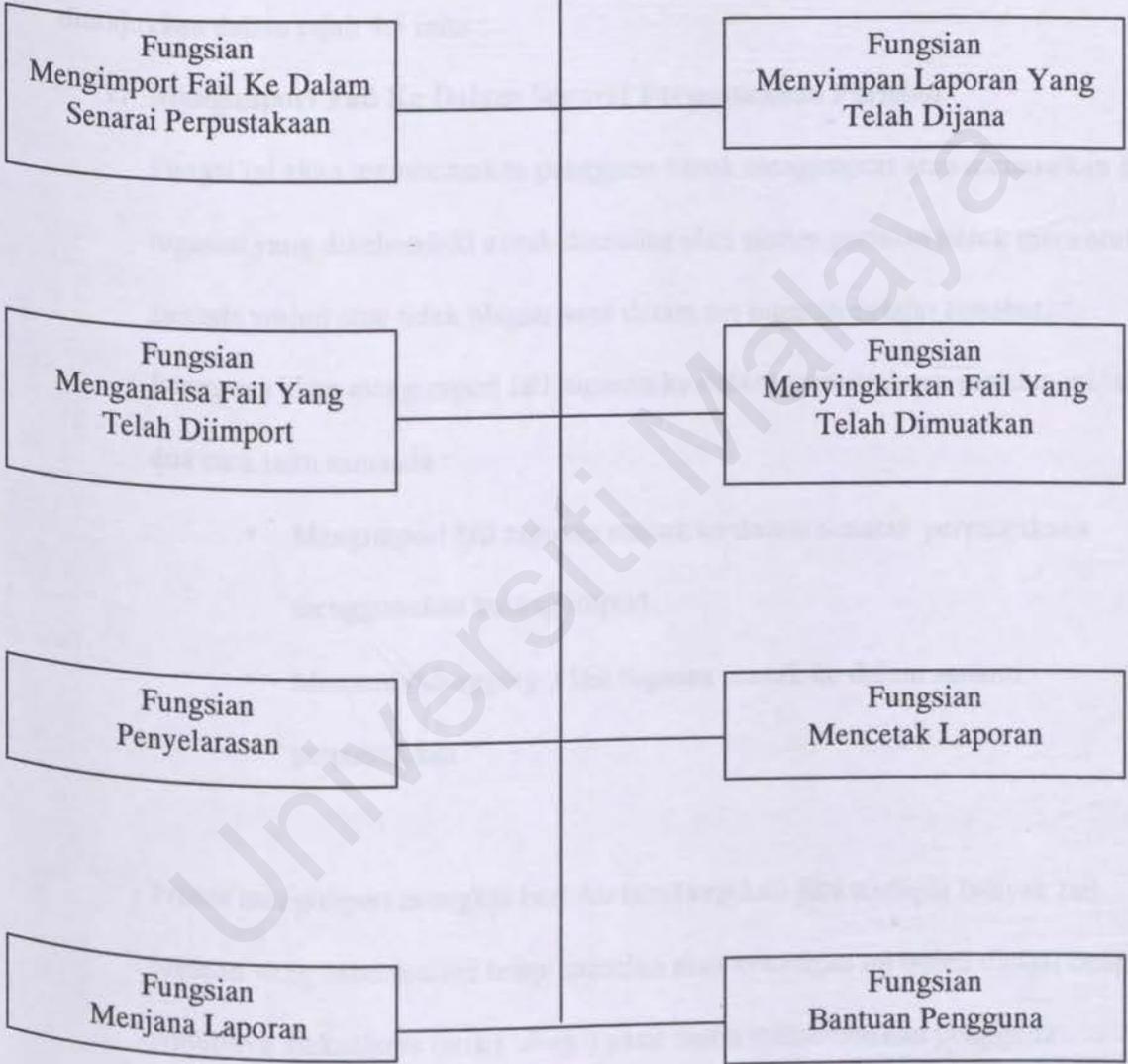
Proses ini amat penting kerana ia membolehkan struktur kandungan pembangunan yang lebih dinamik dengan kewujudan ciri-ciri interaktif. Menurut Ian Sommerville, keperluan boleh dibahagikan kepada 2 iaitu fungsian dan bukan fungsian (Sommerville, 1996).

4.5.1 Keperluan Fungsian

Keperluan fungsian menggariskan fungsi utama sistem. Ia menerangkan interaksi antara sistem dengan persekitarannya dan juga bagaimana sesuatu sistem akan bertindakbalas pada suatu keadaan (Davis, 1993).

Berdasarkan daripada rumusan hasil tinjauan yang telah dibuat, keperluan fungsian ditentukan.

Perisian Penganalisa Plagiarisme



Rajah 4.5 : Pecahan Komponen Fungsian Bagi Perisian Penganalisa Plagiarisme

Rajah 4.5 menunjukkan pecahan struktur fungsian bagi perisian penganalisa plagiarisme yang mana setiap pecahan tersebut akan menerangkan keperluan atau kelakuan bagi sistem perisian tersebut. Secara umumnya, keperluan fungsian perisian penganalisa plagiarisme ini boleh dikelaskan kepada 8 fungsian utama seperti yang ditunjukkan dalam rajah 4.5 iaitu :

1) Mengimport Fail Ke Dalam Senarai Perpustakaan Perisian

Fungsi ini akan membenarkan pengguna untuk mengimport atau memuatkan fail tugas yang dikehendaki untuk dianalisa oleh sistem perisian untuk menentukan samada wujud atau tidak plagiarisme dalam set tugas pelajar tersebut.

Pengguna akan mengimport fail tugas ke dalam perpustakaan perisian melalui dua cara iaitu samaada :

- Mengimport fail tugas masuk ke dalam senarai perpustakaan menggunakan butang import.
- Menarik ('*dragging*') fail tugas masuk ke dalam senarai perpustakaan

Proses mengimport mungkin berlaku berulang-kali jika terdapat banyak fail tugas yang hendak diuji tetapi masalah atau kekangan ini boleh diatasi dengan wujudnya mekanisme tarik ('*drag*') yang mana membenarkan pengguna memuatkan fail dengan cepat.

2) Menganalisa Fail Yang Telah Diimport

Fungsi ini akan menganalisa ke semua fail tugas yang telah diimport dan dimuatkan oleh pengguna ke dalam senarai perpustakaan perisian. Analisa dilakukan oleh sistem perisian untuk menentukan samada fail tugas yang dianalisa terlibat dengan plagiarisme atau tidak berdasarkan algoritma pengiraan analisa yang dibangunkan.

3) Penyelarasan

Fungsi ini merupakan satu fungsi tambahan yang membenarkan pengguna menyelaras peraturan fungsi analisa sistem supaya tahap pengesanan plagiarisme adalah lebih tepat bergantung kepada kesesuaian tugas yang diberi. Penyelarasan peraturan yang boleh dibuat oleh pengguna ialah :

1) Memilih samada untuk mengambilkira atau tidak, kes bagi huruf besar atau kecil dalam tugas semasa proses analisa

Jika pengguna memilih untuk mengambilkira kes bagi huruf besar atau kecil semasa proses analisa, sebagai contohnya jika wujud perkataan "Plagiarisme" dan "plagiarisme", maka sistem akan mengambilkira kes tersebut sebagai 2 perkataan yang berbeza berbanding jika tidak memilih untuk mengambilkira kes bagi huruf besar atau kecil, maka sistem akan mengambilkira kes tersebut sebagai 1 perkataan yang sama.

- 2) **Memilih samada untuk mengambilkira atau tidak, tanda baca dalam tugas semasa proses analisa.**

Tanda baca yang akan diambilkira oleh sistem merangkumi :

, . “ ‘ ; : ? / \ ! { } [] -

- 3) **Memilih samada untuk mengambilkira atau tidak, nombor dalam tugas semasa proses analisa**

Dalam kes perkataan “10-bit” dan “8-bit” sistem akan menganggap perkataan tersebut adalah 2 perkataan yang sama jika pengguna tidak mengambilkira semasa proses analisa

- 4) **Menetapkan bilangan aksara terpendek dalam suatu perkataan di mana jika perkataan itu mempunyai bilangan aksara yang lebih pendek daripada yang ditetapkan proses analisa akan mengabaikan perkataan tersebut.**

Dalam kes perkataan seperti “dan”, “ia”, “ada”, “atau”, “pada” sistem akan mengabaikannya jika penetapan dibuat bahawa perkataan yang lebih pendek daripada 5 harus diabaikan dalam proses analisa. Fungsi ini penting kerana banyak perkataan biasa yang boleh diabaikan dalam kes-kes biasa kerana ia merupakan perbendaharaan kata terkongsi dalam bahasa.

- 5) **Menetapkan bilangan huruf terpanjang dalam suatu perkataan di mana jika perkataan itu mempunyai bilangan**

huruf yang lebih panjang daripada yang ditetapkan proses analisa akan mengabaikan perkataan tersebut.

Fungsi ini amat berguna dalam kes di mana tugas yang mempunyai item-item bukan tekstual merangkumi nama fail, alamat website dan data imej yang mana ia terdiri daripada aksara yang banyak dalam satu-satu perkataan.

Sebagai cadangan, sistem akan secara lalainya akan mengambil nilai di mana :

- Mengambil kira kes bagi huruf besar atau kecil dalam tugas semasa proses analisa
- Tidak mengambil kira tanda baca dalam tugas semasa proses analisa.
- Tidak mengambil kira nombor dalam tugas semasa proses analisa
- Mengabaikan perkataan di mana bilangan aksara terpendek adalah kurang daripada 4 aksara
- Mengabaikan perkataan di mana bilangan aksara terpanjang adalah lebih daripada 20 aksara

4) Menyingkirkan fail yang telah dimuatkan

Fungsi ini membenarkan pengguna untuk menyingkirkan fail tugas yang telah dimuatkan ke dalam perpustakaan perisian. Penyingkiran fail boleh dilakukan melalui 2 cara iaitu :

- Penyingkiran ke semua senarai tugas yang terdapat di dalam perpustakaan perisian
- Penyingkiran senarai tugas yang dipilih oleh pengguna sahaja

5) Menjana laporan

Setelah proses analisa dilakukan oleh sistem perisian ke atas senarai fail tugas yang dimuatkan ke dalam perpustakaan perisian, sistem perisian akan menjana laporan yang menunjukkan samada wujudnya plagiarisme atau tidak dalam set tugas yang telah diuji atau di analisa. Terdapat 2 jenis laporan yang akan dijana oleh sistem iaitu :

- Laporan yang menunjukkan peratusan plagiarisme yang terlibat antara set tugas yang diuji atau dianalisa dalam bentuk jadual
- Laporan yang lebih lengkap yang menunjukkan frasa dan perenggan yang terlibat yang mempunyai ciri-ciri kesamaan dalam set tugas yang diuji

6) Menyimpan laporan yang telah dijana

Pengguna mempunyai pilihan samaada untuk menyimpan atau tidak bukti laporan yang menunjukkan kes plagiarisme antara set tugas yang telah dianalisa yang mana tujuan penyimpanan laporan ini adalah untuk memudahkan pengguna untuk memanggil semula laporan tersebut tanpa perlu menguji semula fail tugas tersebut.

7) Mencetak laporan

Fungsi ini membenarkan pengguna untuk mencetak laporan yang telah dijana oleh sistem bagi memudahkan pengguna meneliti laporan tersebut selain daripada boleh dijadikan bukti bahawa tugas tersebut terlibat dengan plagiarisme.

8) Bantuan pengguna

Fungsi bantuan pengguna bertujuan untuk memudahkan pengguna untuk mengetahui cara-cara untuk menggunakan sistem serta mengetahui apakah fungsi setiap menu atau ikon yang ditunjukkan dalam antaramuka sistem perisian tersebut.

4.5.2 Algoritma Pengiraan

4.5.2.1 Mengira Peratusan Plagiarisme Melalui Kaedah Peratusan Frekuensi

Kesamaan Bilangan Perkataan Antara Dua Tugas

Kaedah ini secara asasnya bertujuan untuk mengira peratusan plagiarisme antara 2 tugas dengan cara mengira peratusan frekuensi bilangan perkataan yang sama antara 2 tugas tersebut. Konsepnya mudah di mana, lebih banyak frekuensi bilangan perkataan yang sama antara 2 tugas, maka lebih tinggi peratusan plagiarisme antara 2 tugas tersebut. Di sini, mekanisme yang digunakan ialah berdasarkan konsep pengiraan analisa 2 tugas demi 2 tugas yang bermaksud sistem akan membuat analisa ke atas 2 fail tugas dahulu dan setelah selesai sistem akan membuat analisa ke atas 2 fail tugas yang lain pula.

Melalui kaedah ini, plagiarisme antara 2 tugas dapat dikenalpasti dalam kes-kes di mana plagiarisme dilakukan dengan cara mengubah struktur posisi atau kedudukan seperti yang dinyatakan di bawah yang mana kes-kes ini merupakan teknik-teknik plagiarisme utama atau yang biasa dilakukan oleh para pelajar dalam menyiapkan tugas mereka dengan masa yang singkat iaitu :

- Mengubah struktur ayat atau perenggan dalam tugas mereka di mana ayat atau perenggan asal diubah kedudukannya berbanding dengan tugas asal
- Melakukan penggabungan dan pengurangan ayat

Kaedah pengiraan peratusan plagiarisme tersebut adalah seperti dinyatakan seperti di bawah:

Diberi 2 tugas X dan Y masing-masing, peratusan plagiarisme dapat dikira berdasarkan frekuensi bilangan perkataan yang sama dengan :

- 1) Mengira frekuensi bilangan perkataan yang sama iaitu n_{XY} antara 2 tugas iaitu tugas X dan tugas Y dan juga jumlah bilangan perkataan bagi setiap tugas X dan Y.
- 2) Untuk mengira peratusan plagiarisme bagi tugas X dan Y masing-masing formula pengiraan adalah seperti di bawah iaitu :

$$\text{Peratusan plagiarisme bagi tugas, \%} = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Dalam Tugas}}$$

Contohnya, untuk mengira:

Peratusan plagiarisme tugas X terhadap tugas Y ialah :

$$\text{Peratusan plagiarisme bagi tugas, \%}_X = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Tugas X}}$$

dan

Peratusan plagiarisme tugas X terhadap tugas Y ialah :

$$\text{Peratusan plagiarisme bagi tugas, } \%_Y = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Tugas Y}}$$

Contoh sampel kes plagiarisme yang berlaku :

Tugas X mempunyai jumlah bilangan perkataan sebanyak 75, 000 patah perkataan dan tugas Y mempunyai jumlah bilangan perkataan sebanyak 50, 000 patah perkataan dan bilangan perkataan yang sama antara dua tugas adalah 42, 000 patah perkataan, maka berdasarkan kes ini :

Peratusan plagiarisme bagi tugas X kepada tugas Y ialah :

$$\frac{42,000}{75,000} * 100 = 56.0\%$$

Peratusan plagiarisme bagi tugas Y kepada tugas X ialah :

$$\frac{42,000}{50,000} * 100 = 84.0\%$$

4.5.2.2 Implementasi Algoritma Dalam Dunia Sebenar

Dalam dunia sebenar, pengguna akan membuat analisa ke atas lebih daripada 2 fail tugas. Di sini akan ditunjukkan bagaimana sistem membuat analisa ke atas 3 fail tugas sebagai sampel contoh iaitu Fail 1, Fail 2 dan Fail 3 berdasarkan kepada mekanisme 2 tugas demi 2 tugas. Langkah-langkah adalah seperti berikut :

- 1) Sistem akan membaca dan mengekstrak maklumat daripada ketiga-tiga fail tugas tersebut bagi mendapatkan maklumat mengenai jumlah bilangan perkataan bagi setiap fail tugas tersebut.
- 2) Kesamaan perkataan antara 2 fail tugas akan didapatkan iaitu merujuk kepada sampel ini sistem akan mencari bilangan kesamaan perkataan bagi 3 pasangan fail iaitu bilangan pasangan tersebut mestilah $[(n(n-1))/2]$ pasangan di mana n adalah bilangan tugas yang hendak diuji iaitu :
 - Pasangan Fail 1 dengan Fail 2
 - Pasangan Fail 1 dengan Fail 3
 - Pasangan Fail 2 dengan Fail 3
- 3) Kemudian sistem akan membuat pengiraan peratusan plagiarisme berdasarkan mekanisme 2 tugas demi 2 tugas untuk menunjukkan peratusan tersebut iaitu dengan menggunakan formula $[n(n-1)]$ untuk menentukan bilangan pasangan fail tugas yang perlu di analisa di mana n merujuk kepada bilangan

tugasan yang hendak diuji, maka dalam konteks ini bilangan pasangan tugas

yang mesti ialah 6 pasang iaitu :

- Fail 1 dengan Fail 2
- Fail 1 dengan Fail 3
- Fail 2 dengan Fail 1
- Fail 2 dengan Fail 3
- Fail 3 dengan Fail 1
- Fail 3 dengan Fail 2

Di mana berdasarkan pasangan ini, sistem akan melaporkan peratusan plagiarisme bagi setiap pasangan fail tugas seperti yang di atas.

4.5.2.3 Menentukan Tahap Peratusan Plagiarisme

Bagi menentukan berapakah peratusan yang boleh dikatakan sesuatu tugas terlibat dengan plagiarisme atau tidak, satu kajian telah dibuat oleh pengarang perisian CopyCatch yang merupakan kumpulan forensik linguistik, Birmingham University yang diketuai oleh Profesor Malcolm Coulthard mengatakan bahawa (Wools, 1999):

- Peratusan kesamaan pertindihan perkataan antara 2 tugas yang ditulis secara bebas oleh 2 pengarang adalah di dalam lingkungan 40 % adalah normal
- Jika peratusan melebihi 70% maka satu kajian lanjut perlu dilakukan ke atas tugas tersebut yang memungkinkan berlakunya plagiarisme antara tugas tersebut.

Berdasarkan kajian disertasi sarjana yang dilakukan oleh Susan Finlay (Finlay, 1999) mengatakan bahawa, 2 tugas yang ditulis secara bebas mempunyai peratusan kesamaan pertindihan perkataan sebanyak 50% atau lebih tetapi penggunaan perkataan tersebut adalah berbeza antara satu pengarang dengan pengarang yang lain.

Oleh itu berdasarkan kepada 2 kajian yang dijalankan di atas saya menetapkan bahawa peratusan plagiarisme antara set tugas adalah seperti dalam jadual 4.1 di bawah iaitu :

Tahap Lingkungan Peratusan	Deskripsi
0% hingga 40%	Merupakan satu peratusan yang bebas daripada plagiarisme
41% hingga 70%	Merupakan satu peratusan yang normal
71% hingga 100%	Merupakan satu peratusan yang memungkinkan berlakunya plagiarisme antara tugas dan kajian terperinci perlu dilakukan ke atas tugas tersebut.

Jadual 4.1: Penentuan Tahap Peratusan Plagiarisme

Tetapi peratusan plagiarisme ini adalah terpulang kepada pengguna iaitu para pensyarah untuk menentukan samada tugas tersebut terlibat atau tidak dengan plagiarisme berdasarkan peratusan yang dijana oleh sistem perisian penganalisa plagiarisme ini.

4.5.3 Keperluan Bukan Fungsian

Keperluan bukan fungsian pula merujuk kepada had-had atau halangan terhadap perkhidmatan-perkhidmatan yang disediakan oleh sistem. Ia juga mengambil kira had masa semasa proses pembangunan sistem.

Berikut merupakan keperluan bukan fungsian sistem yang akan dibangunkan adalah :

1) Ketepatan

Sistem perisian yang akan dibangunkan mesti menjana laporan yang tepat bagi melaporkan kes plagiarisme yang berlaku dalam set tugas yang diuji. Dalam konteks ini, pengiraan algoritma telah dibangunkan serta fungsi penyelarasan nilai lalai ('*default value*') yang terdapat di dalam sistem perisian bagi menjamin ketepatan sistem.

2) Kebolehgunaan

Pengguna mudah memahami dan berinteraksi dengan sistem perisian yang akan dibangunkan serta mudah untuk menggunakan setiap komponen fungsian yang terdapat di dalam perisian tersebut. Dalam konteks ini, apa yang dijangkakan ialah pengguna baru memerlukan masa yang singkat untuk mempelajari bagaimana untuk menggunakan sistem perisian ini dan jangkaan masa yang diperlukan untuk latihan kepada pengguna baru tersebut untuk menggunakan perisian ini adalah 15 minit bagi menjalankan proses dan operasi biasa perisian. Selain daripada itu

juga, antaramuka grafik pengguna adalah mengikut piawaian Microsoft dalam memaparkan tettingkap, borang input pengguna, butang-butang, fail menu dan lain-lain bagi memudahkan pengguna untuk menggunakan perisian ini.

3) Berprestasi tinggi dengan maklumbalas yang minima

Proses analisa dan pengesanan plagiarisme yang dilakukan oleh sistem perisian hendaklah dalam tempoh masa yang munasabah dan boleh diterima oleh pengguna. Dalam konteks ini, masa tindakbalas analisa adalah di dalam purata 2 saat untuk kes perbandingan kapasiti tugas sebanyak 20 fail tugas dengan penggunaan sumber memori yang rendah.

4) Kebolehgunaansemula

Setiap komponen perisian haruslah direkabentuk supaya komponen fungsian tersebut bersifat mudah untuk diperluaskan dan boleh digunakan semula sekiranya berlakunya modifikasi atau pengubahsuaian pada sistem perisian pada masa akan datang. Dalam konteks ini, bagi memudahkan proses penyelenggaraan dan kebolehgunaansemula semula satu piawaian prosedur untuk pengkodan, penamaan pembolehubah, perpustakaan kelas dan utiliti penyelenggaraan dilakukan semasa proses pembangunan.

5) Keefisienan

Keefisienan merujuk kepada keupayaan sistem untuk memproses beberapa fail tugas per unit masa yang munasabah selain daripada meminimumkan penggunaan ruang ingatan dan unit pemrosesan pusat.

Universiti Malaya

4.6 Keperluan Perisian

Seksyen ini menerangkan keperluan perisian semasa proses pembangunan untuk membangunkan perisian penganalisa plagiarisme ini.

1) Bahasa Pengaturcaraan

Untuk membangunkan perisian penganalisa plagiarisme ini, saya telah menggunakan bahasa C++ sebagai bahasa pengaturcaraan pembangunan. Ini disebabkan kerana bahasa C++ merupakan bahasa yang berorientasikan objek dan berkuasa serta memberikan sepenuh kawalan dan kuasa kepada pengaturcara untuk mengawal aliran data input dan output bagi sistem. Penggunaan konsep kelas, objek, enkapsulasi, perwarisan serta polimorfisma akan dioptimumkan penggunaan sepenuhnya bagi memudahkan kerja-kerja membangun dan menyahpejijat projek ini. Selain daripada itu juga, C++ amat sesuai dalam menyelesaikan kerja-kerja pengaturcaraan yang melibatkan kes penyelesaian masalah dan juga cenderung kepada pembangunan algoritma yang lebih mendalam untuk menyelesaikan masalah tersebut. Oleh kerana ia berteraskan kepada penggunaan pengkompil ('*compilers*') ia menjamin prestasi yang tinggi ke atas program yang dilarikan.

2) Alatan Pembangunan

Untuk memudahkan kerja-kerja pengaturcaraan menggunakan bahasa pengaturcaraan C++, alatan pembangunan yang sesuai digunakan ialah Microsoft Visual Studio 6.0 dengan Visual C++. Visual C++ menjadikan proses pembangunan dan pengaturcaraan menjadi cepat dan mudah selain daripada itu juga ia sentiasa mengekalkan ciri C++ yang mana teras kebolehannya berasaskan kepada konsep berorientasikan objek. Visual C++ juga turut menyediakan alatan untuk membina antaramuka grafik pengguna seperti tettingkap, borang input pengguna, butang-butang yang menyokong sistem pengendalian Microsoft Windows.

3) Platform Sistem Pengendalian

Sistem pengendalian yang diperlukan untuk membangunkan dan melarikan perisian ini ialah sistem pengendalian yang berplatformkan Microsoft Windows tidak kira Windows 3.X, Windows 9.X, Windows ME, Windows 2000, Windows NT ataupun Windows XP.

4.7 Keperluan Perisian

Kesemua keperluan perisian diringkaskan dalam jadual yang di bawah :

No.	Peralatan Atau Persekitaran Sistem	Jenis Keperluan Perisian
1.	Bahasa Pengaturcaraan	C++
2.	Alatan Pembangunan	Microsoft Visual C++
3.	Platform Sistem Pengendalian	Platform Windows

Jadual 4.2 : Keperluan Perisian Bagi Membangunkan Perisian Penganalisa

Plagiarisme

4.7 Keperluan Perkakasan

Jadual 4.2 merupakan keperluan perkakasan yang diperlukan pengguna untuk menggunakan dan melarikan perisian penganalisa plagiarisme ini :

No.	Perkakasan	Minima	Cadangan
1.	Unit Pemprosesan Pusat (CPU)	Berasaskan Senibina Pentium (100 MHz)	Berasaskan Senibina Pentium (233Mhz ke atas)
2.	Ruang Cakera Keras	3MB	10MB
3.	Ingatan Capaian Rawak (RAM)	8 MB	32MB
4.	Sistem Pengendalian	Platform Microsoft Windows	Platform Microsoft Windows
5.	Peranti Masukkan	Papan Kekunci Tetikus	Papan Kekunci Tetikus
6.	Peranti Keluaran	Monitor VGA Pencetak Dot Matrik	Monitor Digital Pencetak Deskjet

Jadual 4.3 : Keperluan Perkakasan Pengguna Untuk Melarikan Perisian Penganalisa Plagiarisme Ini.

REKABENTUK SISTEM

5.1 Pengantar

Proses rekabentuk sistem merupakan salah satu fasa yang utama dalam pembangunan sistem di mana keputusan dibuat yang telah ditetapkan dalam fasa sebelumnya ditinjau atau ditukarkan kepada bentuk sistem yang memenuhi keperluan dan keperluan pengguna berdasarkan deskripsi yang telah ditetapkan dan inisiatif yang terdahulu. Secara ringkas, rekabentuk sistem adalah proses yang bertujuan untuk menghasilkan perincian kepada satu bentuk perisian (Purba, 1990). Ia juga merangkumi **REKABENTUK SISTEM** yang diperlukan untuk memastikan bahawa rekabentuk sistem yang dihasilkan memenuhi keperluan yang ditetapkan.

BAB 5

REKABENTUK SISTEM

Rekabentuk sistem adalah langkah kepada dua fasa :

1) Rekabentuk Logikal

Ia merupakan rekabentuk kerangka kepada perisian sistem yang akan dihasilkan yang dibina oleh sistem yang telah direka bentuk. Rekabentuk logikal ini merangkumi bagaimana fungsinya dan terbahagi kepada beberapa bahagian dan bahagian.

REKABENTUK SISTEM

5.1 Pengenalan

Fasa rekabentuk sistem merupakan salah satu fasa yang utama dalam pembangunan sistem di mana keperluan sistem yang telah ditentukan dalam fasa sebelumnya diterjemah atau ditukarkan kepada ciri-ciri sistem yang memenuhi keperluan dan kepuasan pengguna merangkumi deskripsi lengkap bagi fungsi dan interaksi yang terlibat. Secara ringkasnya, ia boleh ditakrifkan sebagai satu proses kreatif bagi menukarkan permasalahan kepada satu bentuk penyelesaian (Pfleeger, 1998). Ia juga merangkumi penggunaan maklumat daripada spesifikasi keperluan untuk menerangkan masalah dan suatu penyelesaian akan diberikan sekiranya ia memenuhi keseluruhan spesifikasi keperluan.

Rekabentuk sistem terbahagi kepada dua iaitu :

1) **Rekabentuk Logikal**

Ia merupakan rekabentuk terancang kepada pengguna tentang perkara yang dilakukan oleh sistem yang mana rekabentuk logikal ini menyatakan bagaimana fungsian dan teras bagi sistem ini berfungsi dan bergerak.

2) Rekabentuk Antaramuka

Merujuk kepada rekabentuk yang menerangkan kefahaman pembangun sistem atau perisian tersebut yang mana berdasarkan kepada rekabentuk antaramuka ini pengguna akan menilai dan mengetahui bagaimana sistem perisian tersebut berfungsi seperti perisian yang sebenar berdasarkan kepada rekabentuk logikal yang telah dibuat.

5.2 Rekabentuk Logikal Sistem Perisian

Satu jujukan arahan-arahan dipanggil algoritma. Algoritma merupakan asas kepada sebahagian pengiraan dalam bidang komputer di mana algoritma digunakan sebagai binaan asas kepada proses-proses yang hendak dilakukan. Maka oleh itu, terdapat 2 alatan atau kaedah piawai yang telah dibangunkan untuk membantu mendokumentasikan algoritma iaitu carta alir dan kod-pseudo. Kebiasaannya carta alir digunakan untuk masalah yang kecil berbanding kod-pseudo sesuai digunakan untuk masalah yang lebih besar (Lipshutz, 2000).

Maka, dalam konteks untuk membangunkan rekabentuk logikal bagi pembangunan perisian penganalisa plagiarisme ini, saya telah menggunakan carta alir atas tujuan yang tersebut.

5.2.1 Carta Alir

Carta alir merupakan paparan visual bagi menunjukkan langkah berjujukan yang diperlukan untuk melaksanakan sesuatu proses daripada awal hingga akhir. Penggunaan simbol-simbol yang mempunyai deskripsi mengenai langkah dalam sesuatu proses yang mana simbol-simbol ini digabungkan secara rangkaian berdasarkan kepada aktiviti sebelum dan selepas memudahkan untuk menggambarkan bagaimana proses itu dilakukan.

Carta alir membantu proses pembangunan untuk mengenalpasti, mengumpul dan menganalisa setiap langkah dalam kes pembangunan satu sistem yang berkonsepkan kepada proses penyelesaian masalah yang mana ia bertujuan untuk menakrifkan dengan lebih jelas setiap proses yang telah dikenal pasti.

Antara kebaikan menggunakan carta alir ialah :

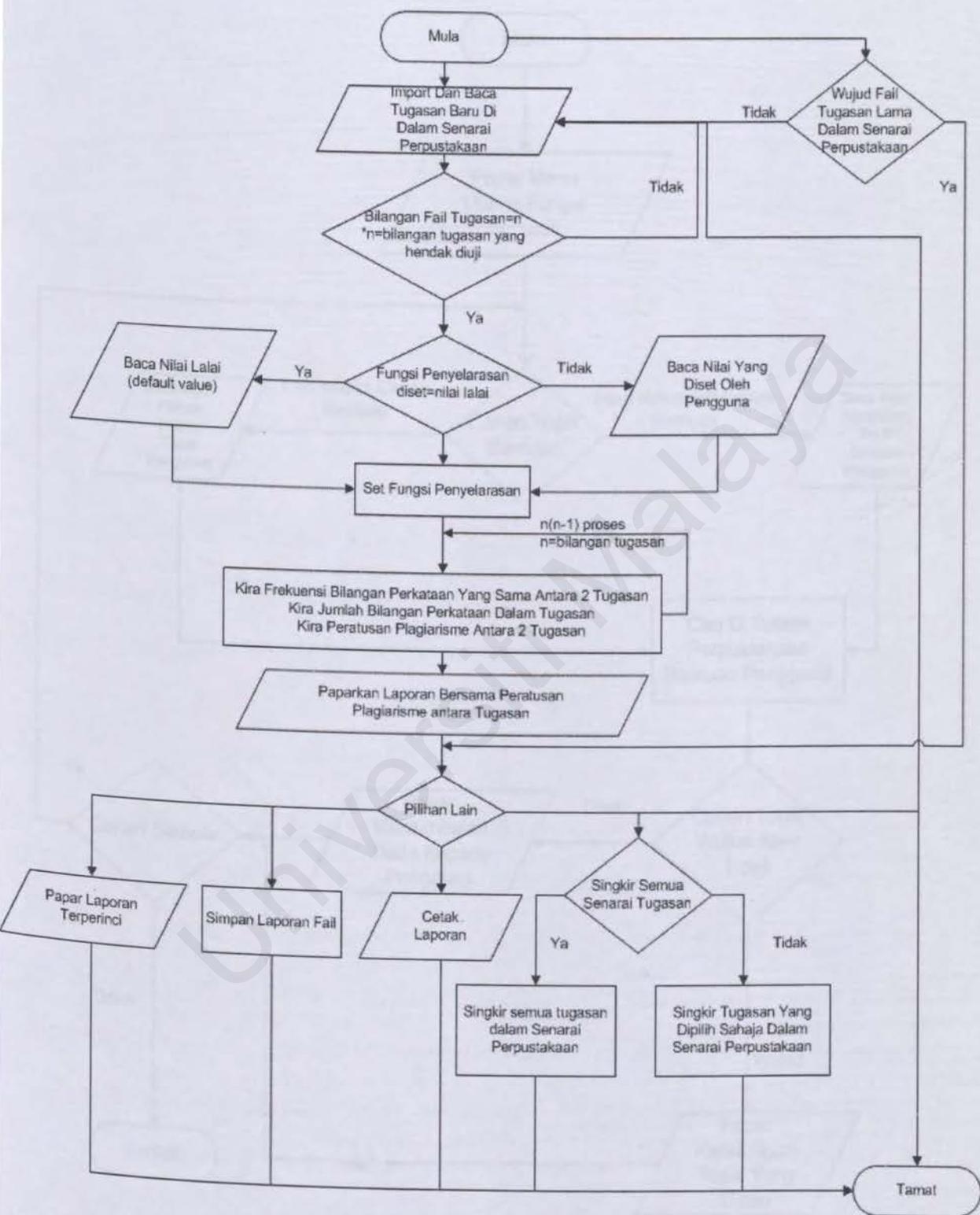
- Paparan visual membantu untuk memudahkan pembangun, pelanggan dan pengguna memahami bagaimana sistem itu beroperasi berdasarkan kepada setiap jujukan langkah dan proses terperinci secara menyeluruh.
- Membantu menggambarkan skop bagi proses-proses pembangunan sistem tersebut.
- Merupakan satu alatan yang mudah digunakan dan diaplikasikan kepada kes yang berkonsepkan kepada kes penyelesaian masalah
- Penggunaan kaedah kod-pseudo adalah rumit dan sukar ditafsirkan secara keseluruhannya berbanding kaedah carta alir kerana ia melibatkan penggunaan kod-kod umum aturcara seperti IF, FOR, WHILE, INPUT dan lain-lain lagi dalam memaparkan rekabentuk logikal sistem.

Rajah 5.1 merupakan carta alir yang menunjukkan bagaimana jujukan langkah bermula daripada pengguna :

- Mengimport fail tugas ke dalam senarai perpustakaan sistem yang mana proses ini berlaku berulang kali sehingga ke semua tugas yang hendak diuji dimuatkan dan seterusnya sistem akan membaca fail tugas tersebut

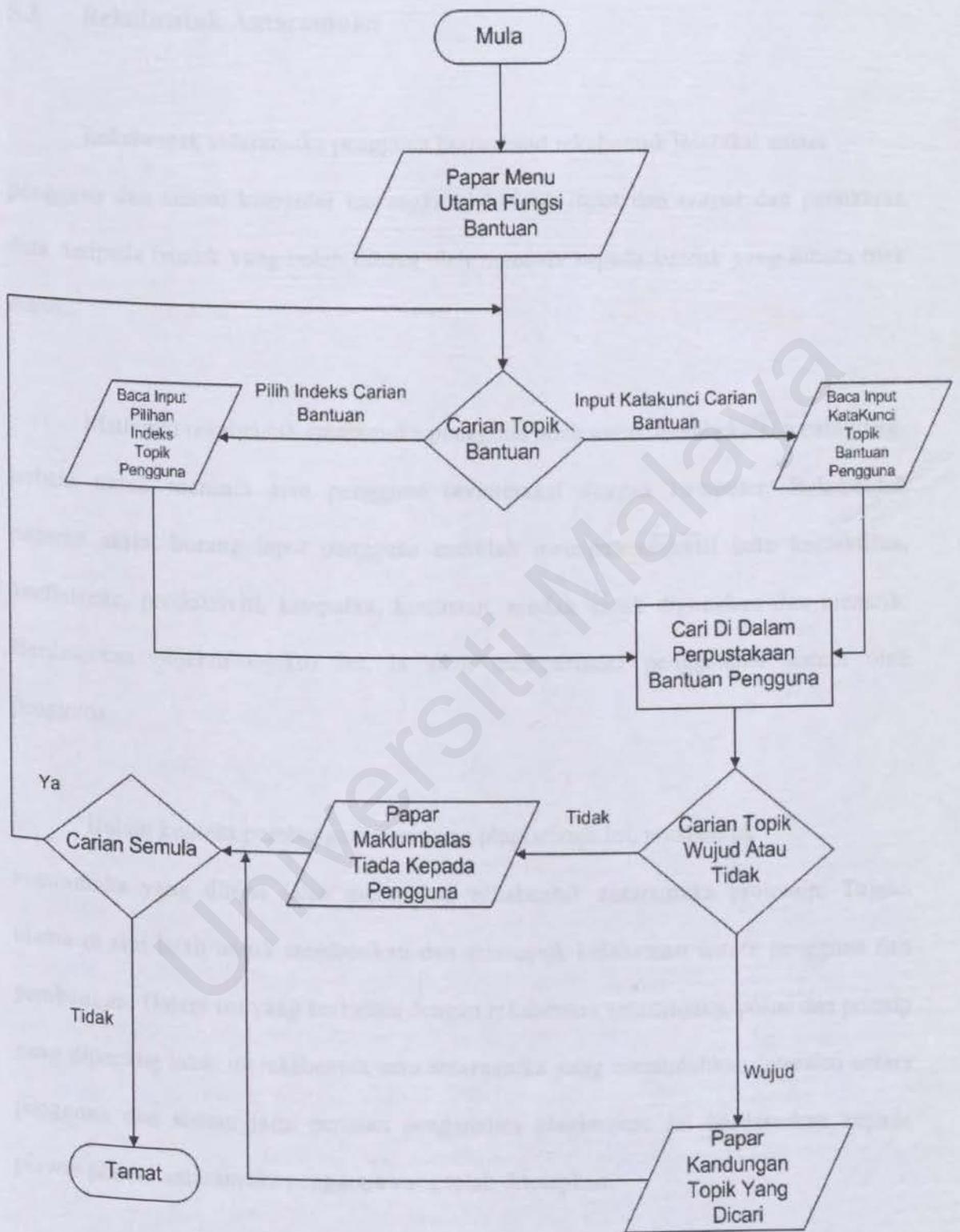
- Setelah itu sistem akan membaca fungsi penyelarasan di mana jika pengguna tidak menetapkan fungsi penyelarasan tersebut sistem akan mengambil nilai lalai ('*default value*') yang telah ditetapkan oleh sistem.
- Setelah itu, sistem akan membuat kiraan peratusan plagiarisme ke atas fail-fail tugas berdasarkan formula yang telah ditetapkan sehingga ke semua fail tugas siap dianalisa.
- Setelah itu pengguna boleh melihat paparan laporan, paparan laporan terperinci, menyimpan laporan, menyingkirkan semua atau sebahagian senarai perpustakaan atau mencetak laporan yang dijana

Rajah 5.2 merupakan carta alir yang menunjukkan bagaimana pengguna boleh mendapatkan topik bantuan bagaimana untuk menggunakan perisian berdasarkan kepada katakunci yang dimasukkan atau senarai indeks pilihan yang boleh dipilih pengguna dan setelah itu sistem akan memaparkan antaramuka maklumat bantuan yang dikehendaki oleh pengguna.



Rajah 5.1 : Carta Alir Fungsi Mengimport Fail Tugas Hingga Penjanaan

Laporan Plagiarisme



Rajah 5.2 : Carta Alir Fungsi Bantuan Pengguna Bagi Perisian Penganalisa

Plagiarisme

5.3 Rekabentuk Antaramuka

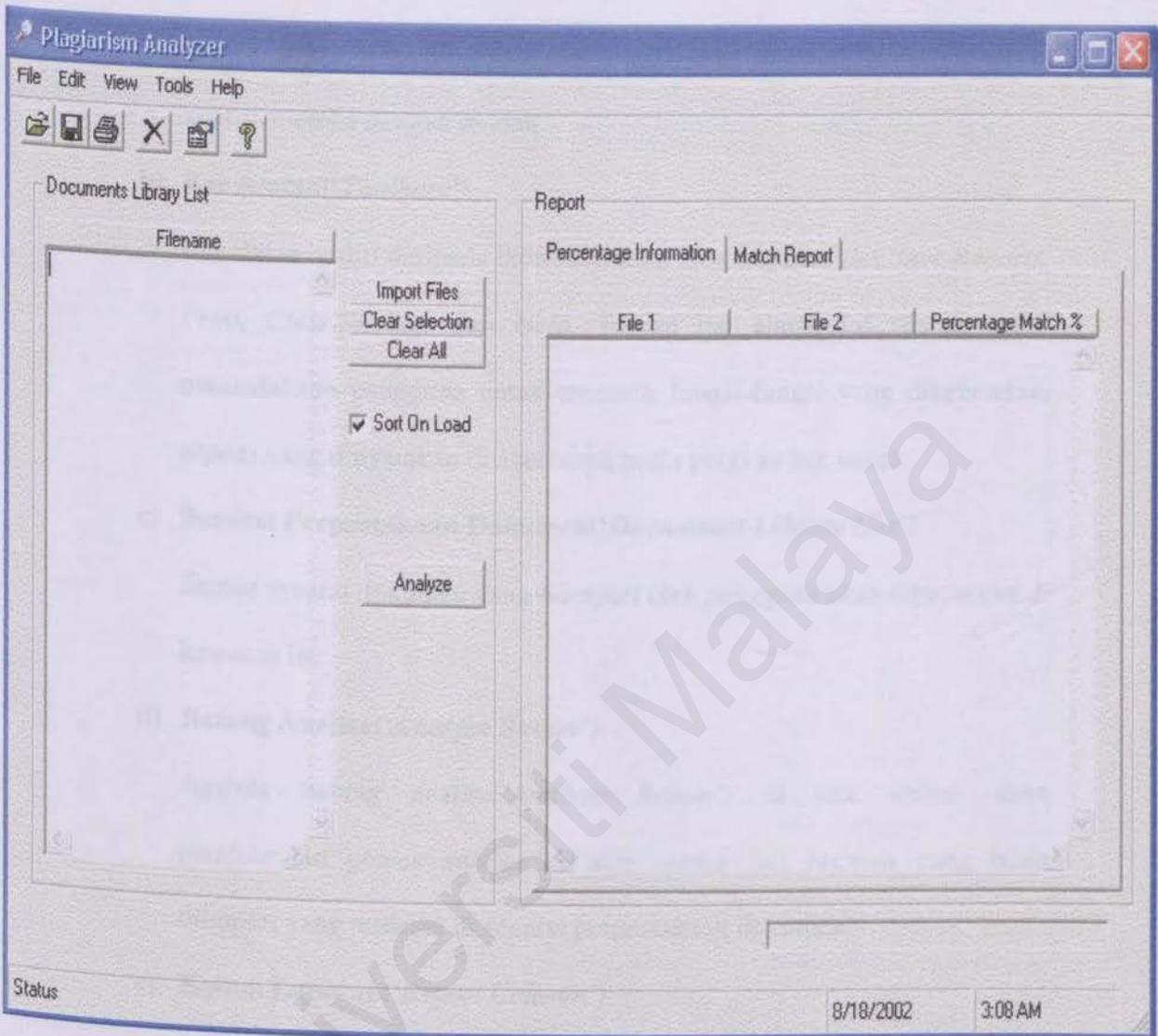
Rekabentuk antaramuka pengguna bermaksud rekabentuk interaksi antara pengguna dan sistem komputer merangkumi kaedah input dan output dan penukaran data daripada bentuk yang boleh dibaca oleh manusia kepada bentuk yang dibaca oleh mesin.

Matlamat rekabentuk antaramuka pengguna ialah untuk membekalkan cara yang terbaik untuk manusia atau pengguna berinteraksi dengan komputer. Rekabentuk paparan skrin, borang input pengguna mestilah menepati objektif iaitu keefektifan, keefisienan, produktiviti, ketepatan, konsisten, mudah untuk digunakan dan menarik. Berdasarkan objektif-objektif ini, ia akan memperbaiki penggunaan sistem oleh pengguna.

Dalam konteks pembangunan perisian plagiarisme ini, rekabentuk antaramuka yang dibina ialah merupakan rekabentuk antaramuka prototaip. Tujuan utama di sini ialah untuk memberikan dan memupuk kefahaman antara pengguna dan pembangun. Dalam isu yang berkaitan dengan rekabentuk antaramuka, polisi dan prinsip yang dipegang ialah merekabentuk satu antaramuka yang memudahkan interaksi antara pengguna dan sistem iaitu perisian penganalisa plagiarisme ini berdasarkan kepada piawai-piawai antaramuka pengguna yang telah ditetapkan.

Rajah-rajah di bawah menunjukkan antaramuka untuk perisian penganalisa

plagiarisme:



Rajah 5.3 : Prototaip Antaramuka Utama Perisian Penganalisa Plagiarisme

Merujuk kepada rajah 5.3, menunjukkan rekabentuk prototaip antaramuka utama perisian penganalisa plagiarisme yang mana ia terdiri daripada :

a) Bar Menu ('Menubars')

Fungsi keseluruhan sistem dikelaskan kepada menu-menu yang berkaitan iaitu menu *File*, *Edit*, *View*, *Tools* dan *Help* yang mana ia bertujuan untuk

membenarkan pengguna memilih fungsi yang ditawarkan oleh perisian kepada mereka dengan mudah.

b) Bar Alatan('Toolbars')

Bar alatan terdiri daripada ikon-ikon iaitu ikon *Import Files, Save Reports, Print, Clear, Settings dan Help*. Tujuan bar alatan ini adalah untuk memudahkan pengguna untuk memilih fungsi-fungsi yang dikehendaki seperti yang dinyatakan di atas tanpa perlu pergi ke bar menu.

c) Senarai Perpustakaan Dokumen('Documents Library List')

Semua senarai dokumen yang diimport oleh pengguna akan dipaparkan di kawasan ini.

d) Butang Analisa('Analyze Button')

Apabila butang analisa('Analyze Button') di klik sistem akan menjalankan proses analisa ke atas semua fail tugas yang telah diimport yang terdapat di senarai perpustakaan dokumen.

e) Kolum Laporan('Report Column')

Kolum laporan akan memaparkan peratusan plagiarisme antara fail tugas yang telah diuji.

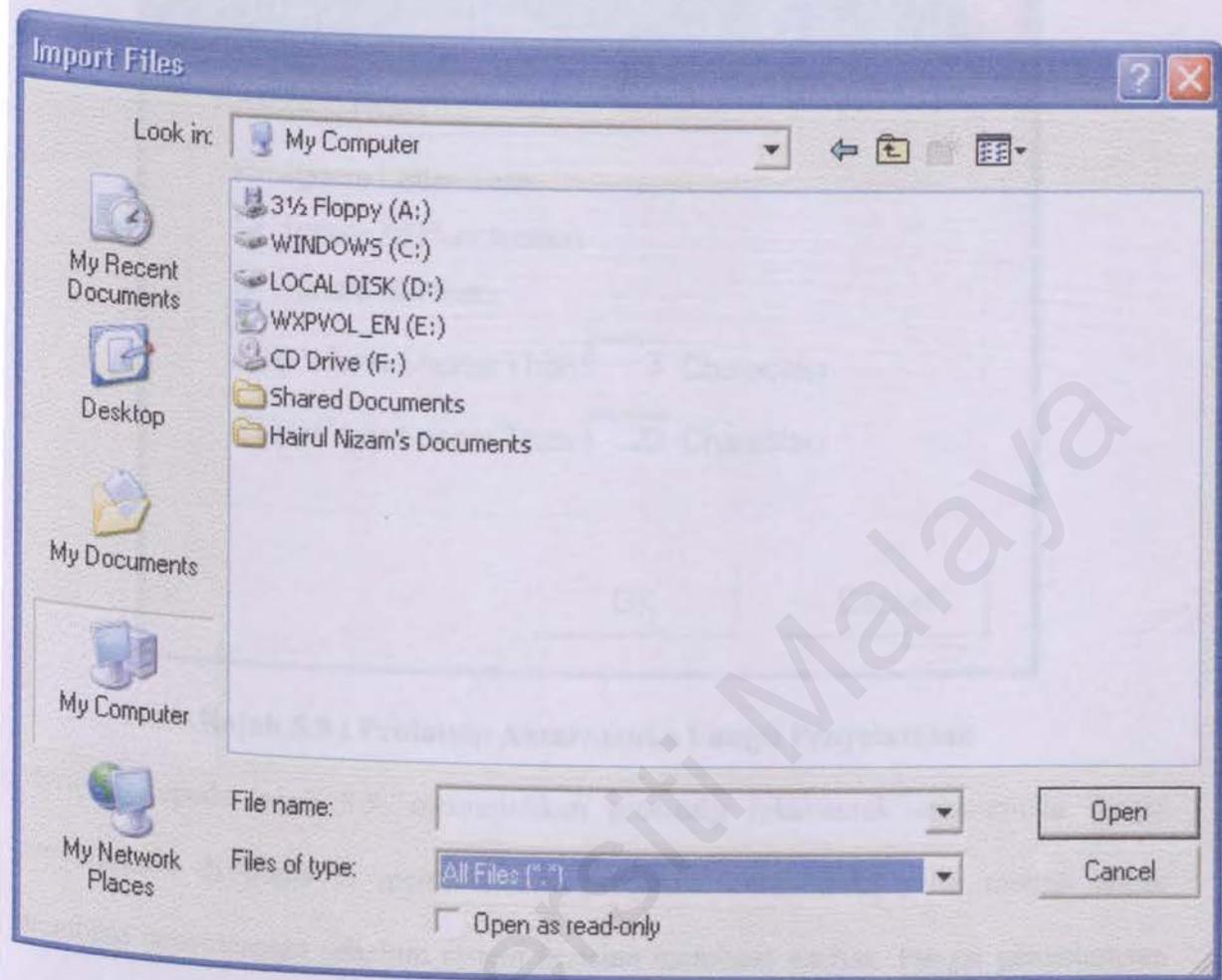
f) Bar Status('Status Bars')

Bar status akan memaparkan status sistem perisian samaada status bersedia atau status sedang menganalisa.

g) Bar Kemajuan('Progress Bars')

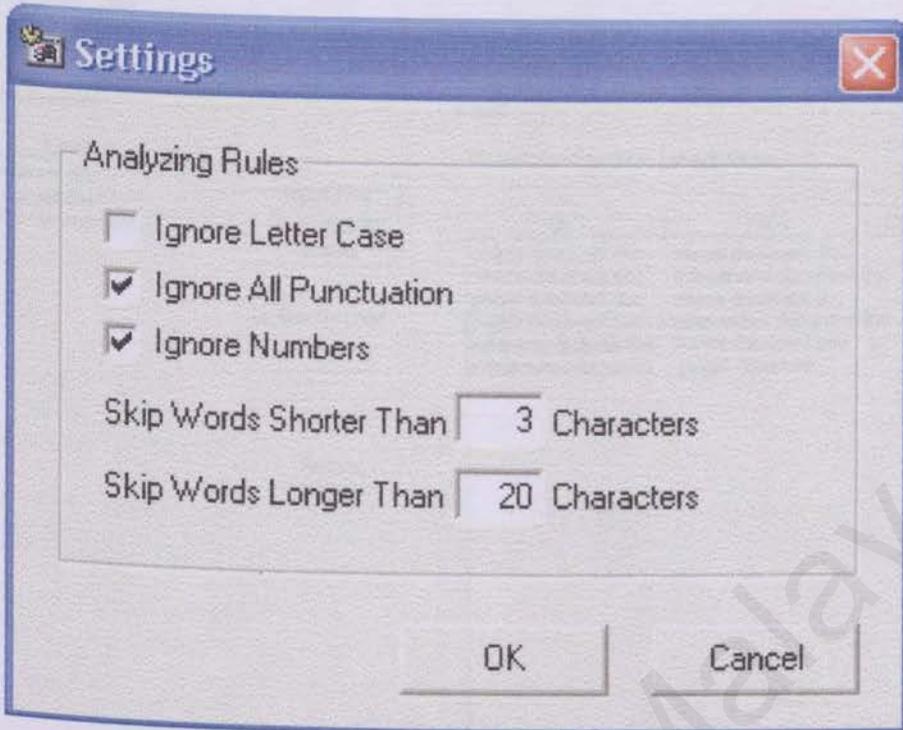
Bar kemajuan akan memaparkan kemajuan semasa sistem menganalisa fail tugas dan ia amat penting supaya pengguna dapat mengetahui kemajuan semasa proses analisa dan menandakan bahawa sistem sedang

menganalisa fail tugasan



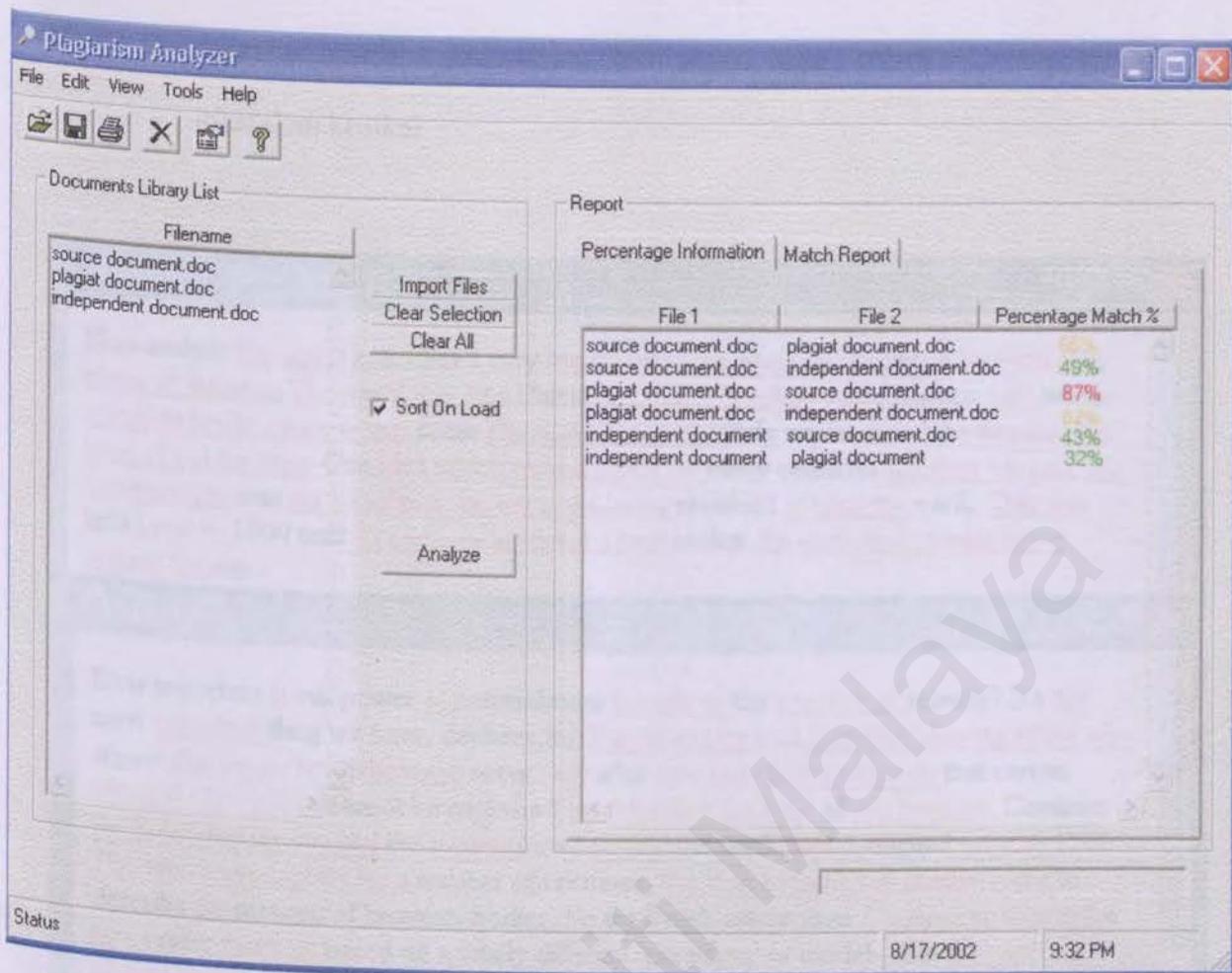
Rajah 5.4 : Prototaip Antaramuka Import Fail

Merujuk kepada rajah 5.4, menunjukkan prototaip rekabentuk antaramuka import fail di mana ia akan dipaparkan apabila pengguna klik pada ikon import fail atau memilih menu import fail untuk memuatkan fail tugasan ke dalam senarai perpustakaan dokumen. Antaramuka ini membenarkan pengguna untuk memilih fail tugasan yang hendak diuji daripada komputer pengguna atau sumber-sumber peranti yang lain dengan mudah.



Rajah 5.5 : Prototaip Antaramuka Fungsi Penyelarasan

Merujuk kepada rajah 5.5, menunjukkan prototaip rekabentuk antaramuka fungsi penyelarasan di mana ia membekalkan pengguna antaramuka yang mudah untuk membuat penyelarasan sebelum sistem perisian membuat analisa. Fungsi penyelarasan seperti yang telah dinyatakan dalam seksyen keperluan fungsian sistem adalah bertujuan untuk mendapatkan laporan yang lebih tepat dan fleksibel berdasarkan kepada kes atau keadaan yang tertentu.

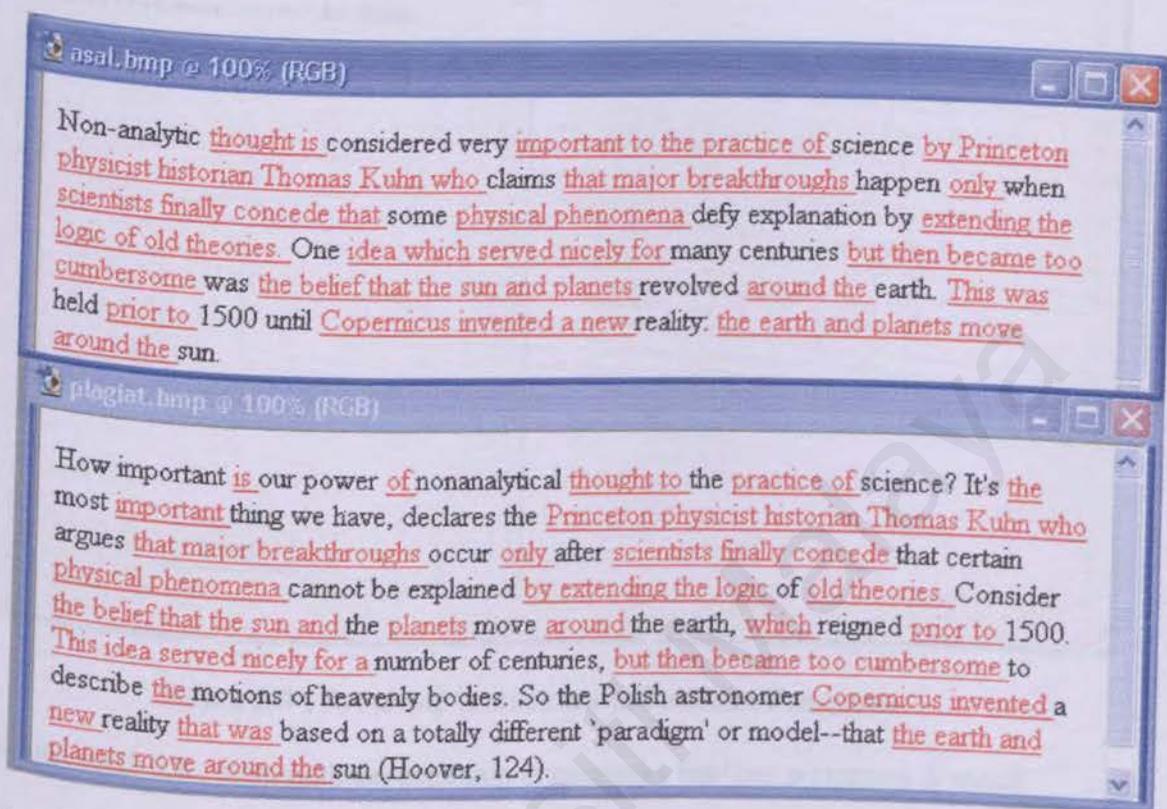


Rajah 5.6 : Prototaip Antaramuka Keseketikaan Yang Menunjukkan Laporan Peratusan Plagiarisme

Merujuk kepada rajah 5.6, menunjukkan prototaip antaramuka keseketikaan yang menunjukkan laporan yang akan di jana dan dipaparkan dalam bentuk peratusan plagiarisme antara set tugas yang diuji. Di sini tulisan peratusan yang berwarna :

- **Hijau** menandakan satu peratusan yang bebas daripada plagiarisme.
- **Kuning** menandakan satu peratusan yang normal tetapi plagiarisme mungkin berlaku dalam kes ini
- **Merah** menandakan satu peratusan yang memungkinkan berlakunya plagiarisme antara tugas dan kajian terperinci perlu dilakukan ke atas

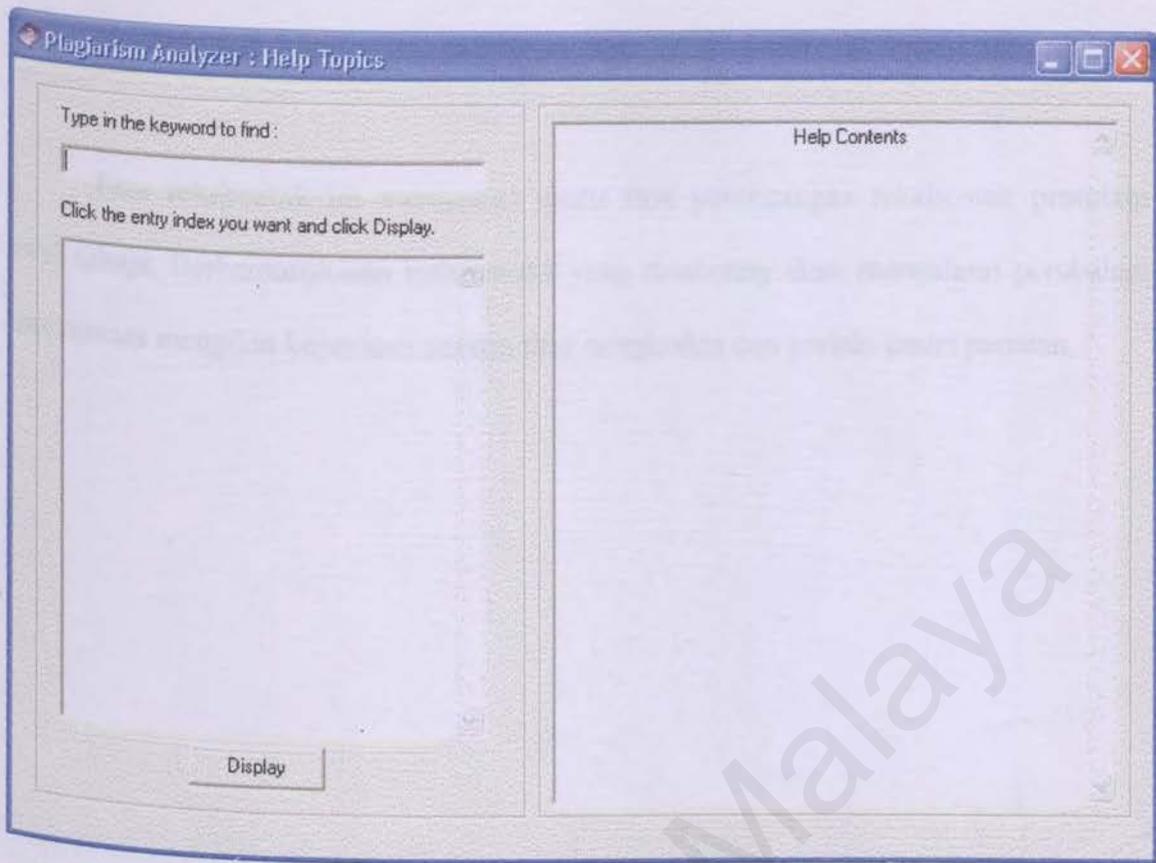
tugas tersebut kerana ini merupakan suatu peratusan yang boleh dikatakan kritikal



Rajah 5.7 : Prototaip Antaramuka Keseketikaan Yang Menunjukkan Laporan

Terperinci Kes Plagiarisme Antara 2 Fail Tugas

Merujuk rajah 5.7, menunjukkan prototaip rekabentuk antaramuka keseketikaan yang memaparkan laporan yang lebih terperinci bagi setiap kes plagiarisme yang berlaku. Sistem akan menggariskan dan menandakan dengan warna merah bagi setiap perkataan yang sama antara set tugas yang diuji bagi menunjukkan dengan jelas kesamaan perkataan atau frasa antara keduanya.



Rajah 5.8 : Prototaip Antaramuka Bantuan Pengguna

Merujuk rajah 5.8, menunjukkan prototaip antaramuka bantuan pengguna di mana jika pengguna memerlukan bantuan untuk menggunakan sistem, pengguna hanya perlu klik pada butang bantu ('*help button*') atau pilih menu bantu. Rekabentuk ini membenarkan pengguna menginput kata kunci pada medan input untuk mendapatkan sebarang bantuan dari sistem yang relevan dengan kata kunci tersebut. Selain daripada itu juga pengguna boleh memilih topik bantuan atau indeks yang dikehendaki olehnya dan seterusnya sistem akan memaparkan sebarang keputusan dan bantuan pada ruangan atau kolum sebelah kanan antaramuka.

5.4 Kesimpulan

Fasa rekabentuk ini merupakan suatu fasa perancangan rekabentuk prototaip awal sahaja. Berkemungkinan antaramuka yang dirancang akan mengalami perubahan dari semasa mengikut keperluan semasa fasa pengkodan dan pelaksanaan perisian.

BAB 6

PENGGODAN DAN BERLAKSANAAN

Universiti Malaya

PENKODAN DAN IMPLEMENTASI

6.1. Pengantar

Para pengkodean dan implementasi adalah pemrograman untuk membuat kode di mana spesifikasi rekayasa dan kebutuhan seperti bahasa pemrograman. Ini termasuk cara apa untuk menulis yang penting untuk mengorganisir dan menganalisis program dari kebutuhan teknis yang akan dirancang. Spesifikasi yang tidak dipatuhi. Tujuan utama dari pemrograman dan implementasi ini adalah untuk menghasilkan kode sumber yang dapat dieksekusi yang memodifikasi

BAB 6

PENKODAN DAN PERLAKSANAAN

6.2. Persekitaran Pengembangan

Sebelum yang dapat dilakukan oleh para ahli komputer dalam, para pengkodean dan implementasi 2003 di mana mereka menggunakan bahasa pemrograman objek. Para ahli komputer menggunakan bahasa Microsoft Visual C++ 5.0. Sedangkan implementasi adalah bahasa pemrograman yang digunakan. Ya atau sebaliknya pengembangan bisa untuk proses pemrograman seperti dalam bentuk di bawah.

PENKODAN DAN IMPLEMENTASI

6.1 Pengenalan

Fasa pengkodan dan implementasi dalam pembangunan sistem merupakan fasa di mana spesifikasi rekabentuk diterjemahkan kepada bahasa pengaturcaraan. Ini merupakan suatu aspek teknikal yang penting untuk mengembangkan perincian spesifikasi perisian dari rekabentuk lakaran yang telah dirancang kepada aplikasi yang boleh dijalankan. Tujuan utama fasa pengkodan dan implementasi ini ialah untuk menghasilkan kod sumber yang jelas, mudah berserta dokumentasi yang memudahkan proses pengujian, pengubahsuaian dan penyelenggaraan pada masa akan datang.

6.2 Persekitaran Pembangunan

Seperti yang dirancang dalam fasa analisa keperluan sistem, teras pengaturcaraan Plagiarizer 2003 dibangunkan menggunakan bahasa berorientasikan objek iaitu C++ menggunakan alatan **Microsoft Visual C++ 6.0**. Walaubagaimanapun terdapat sedikit penambahan yang dilakukan ke atas persekitaran pembangunan iaitu untuk proses pemasangan seperti dalam jadual di bawah :

No.	Deskripsi	Bahasa Pengaturcaraan	Alatan
1.	Teras fungsian pengaturcaraan Plagiarizer 2003 merangkumi : <ul style="list-style-type: none"> • Enjin analisa pengesanan plagiarisme • Pembinaan antaramuka • Penjanaan dan penghasilan laporan 	C++	Microsoft Visual C++ 6.0
2.	Pemasangan Plagiarizer 2003	InstallScript	Install Shield for Microsoft Visual C++

Jadual 6.1 : Persekitaran Pembangunan Plagiarizer 2003

6.3 Fasa Pengkodan Plagiarizer 2003

Dalam fasa pengkodan Plagiarizer 2003, teras fungsian diaturcara menggunakan bahasa C++ yang merangkumi komponen:

1) Enjin analisa pengesanan plagiarisme

Komponen ini merupakan komponen yang terpenting sekali dalam pengaturcaraan dan pembangunan Plagiarizer 2003 kerana komponen ini yang akan melaksanakan enjin analisa untuk mengesan kes plagiarisme dalam tugas pelajar. Selain itu juga, komponen ini merupakan teras untuk memproses ke

semua nilai yang dihantar oleh kelas komponen antaramuka kepadanya. Komponen kelas tersebut ialah kelas CXdocument yang merangkumi fail implementasi doc.cpp dan fail pengepala doc.h.

2) Pembinaan antaramuka

Komponen antaramuka Plagiarizer 2003 juga dibangunkan menggunakan Microsoft Visual C++. Komponen ini merupakan komponen penting untuk mengambil nilai input oleh pengguna seperti nama fail dokumen yang hendak di analisa, aturan penyelarasan yang hendak dilakukan, mengendalikan segala peristiwa(*event*) dan ralat yang dilakukan oleh pengguna seperti klik pada butang dan menghantarkan nilai input tersebut pada komponen enjin analisa serta memaparkan output yang telah diproses oleh enjin analisa Plagiarizer 2003.

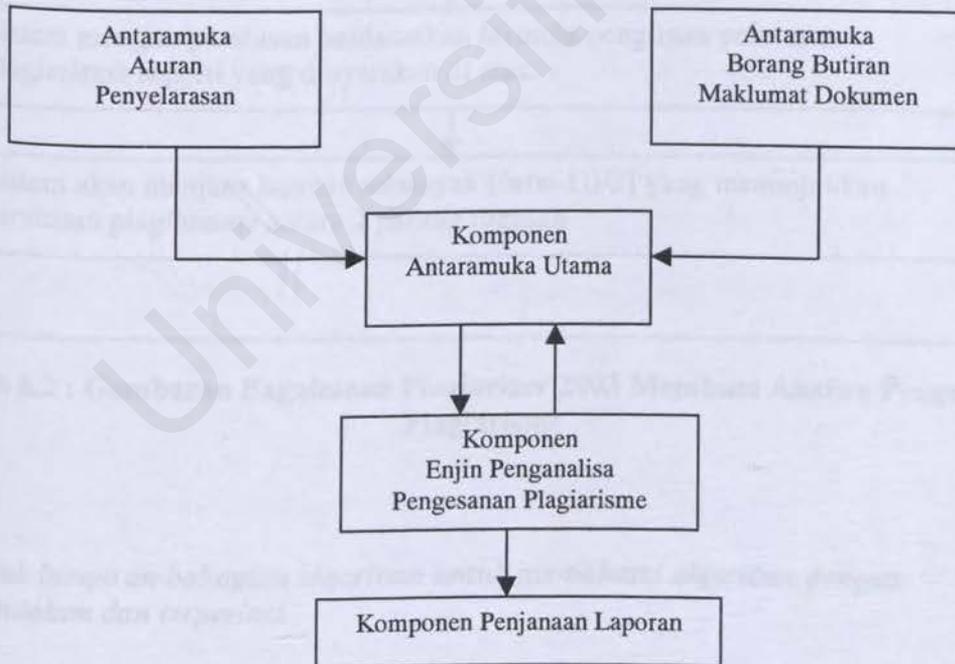
Komponen kelas yang terlibat ialah :

- 1) **CPrototaip2Dlg** – Mengawal antaramuka utama Plagiarizer 2003 yang merangkumi fail prototaip2Dlg.cpp dan fail pengepala prototaip2Dlg.h
- 2) **CSettingDlg** – Mengawal antaramuka aturan penyelarasan Plagiarizer 2003 yang merangkumi fail SettingDlg.cpp dan fail pengepala SettingDlg.h
- 3) **CFormDlg** – Mengawal antaramuka borang butiran maklumat dokumen Plagiarizer 2003 yang merangkumi fail FormDlg.cpp dan fail pengepala FormDlg.h

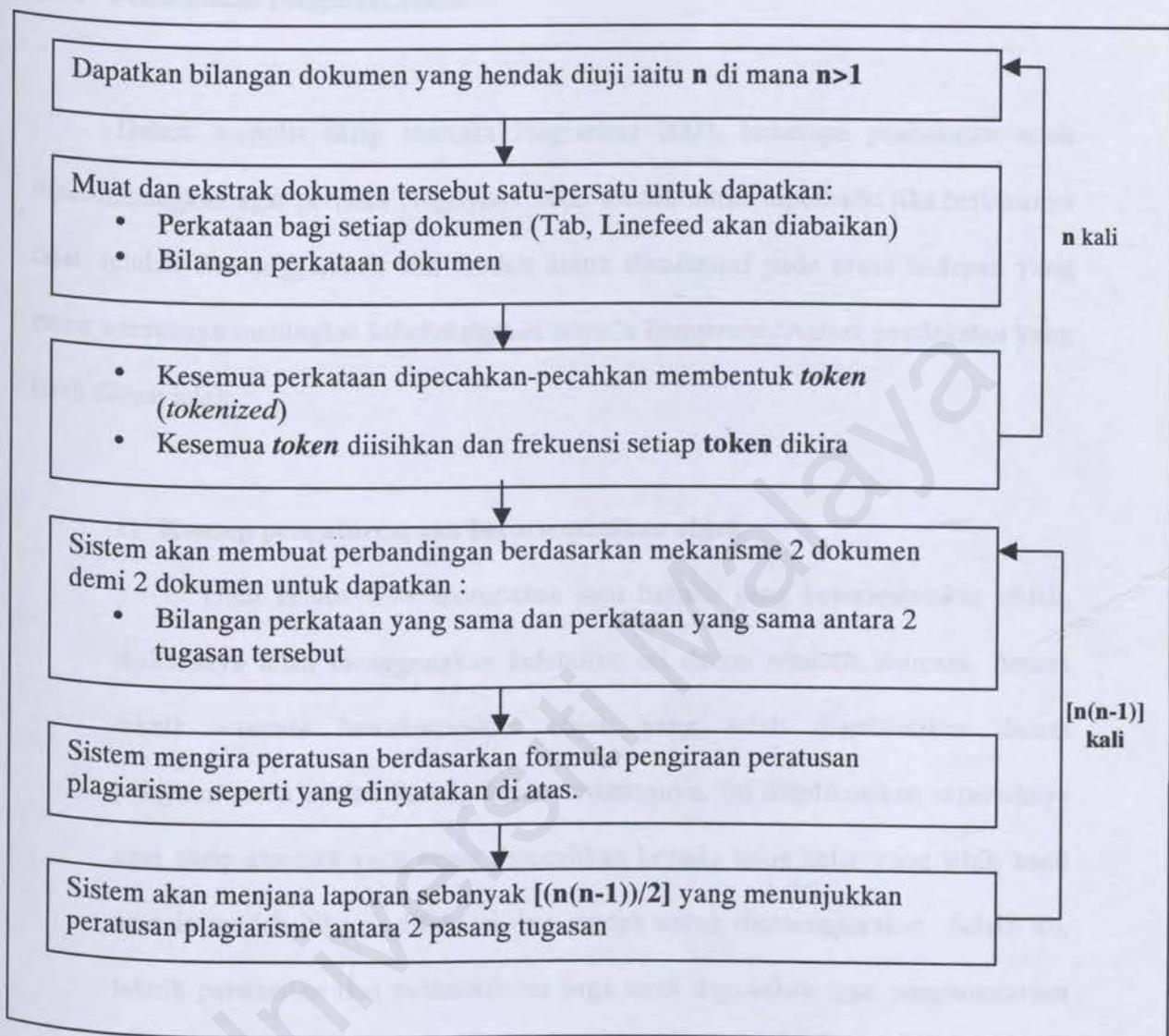
Ketiga-tiga kelas ini disepadukan untuk membentuk satu antaramuka perisian utama yang boleh bekerjasama dengan baik untuk menghantar nilai masing-masing kepada komponen enjin analisa pengesanan plagiarisme.

3) Penjanaan dan penghasilan laporan

Komponen ini digunakan untuk menghasilkan laporan fail HTML yang menunjukkan peratusan plagiarisme setelah proses analisa dilakukan ke atas dokumen oleh kelas komponen enjin analisa pengesanan plagiarisme. Komponen kelas yang terlibat ialah kelas CXdocument yang merangkumi fail implementasi doc.cpp dan fail pengepala doc.h.



Rajah 6.1 : Gambaran Interaksi Antara Komponen Kelas Plagiarizer 2003



Rajah 6.2 : Gambaran Bagaimana Plagiarizer 2003 Membuat Analisa Pengesanan Plagiarisme

*Sila rujuk lampiran bahagian algoritma untuk memahami algoritma dengan lebih mendalam dan terperinci

6.3.1 Pendekatan Pengaturcaraan

Dalam menulis skrip aturcara Plagiarizer 2003, beberapa pendekatan telah dipertimbangkan agar perisian Plagiarizer 2003 mudah untuk diperbaiki jika berlakunya ralat, mudah diselenggarakan dan mudah untuk dinaiktaraf pada masa hadapan yang mana seterusnya meningkat kebolegunaan semula komponen. Antara pendekatan yang telah dibuat ialah:

1) Konsep pengaturcaraan berorientasikan objek

Oleh kerana C++ merupakan satu bahasa yang berorientasikan objek, maka saya telah menggunakan kelebihan ini dalam menulis aturcara. Antara teknik aturcara berorientasikan objek yang telah diaplikasikan dalam pengaturcaraan adalah konsep kelas terutamanya. Ini diaplikasikan sepenuhnya agar skrip aturcara yang besar dipecahkan kepada kelas-kelas yang lebih kecil agar ia mudah dibaca, difahami dan mudah untuk diselenggarakan. Selain itu, teknik perwarisan dan polimorfisma juga turut digunakan agar pengaturcaraan yang dilakukan lebih mudah dan cepat.

2) Dokumentasi dalaman

Untuk dokumentasi dalaman 3 perkara penting telah dititikberatkan dalam melakukan pengaturcaraan Plagiarizer 2003 ini. 3 perkara yang tersebut ialah :

1. Komen pengepala(Header Comment)

Setiap kelas yang dicipta di letakkan komen pengepala agar peranan dan fungsi setiap kelas diketahui. Contoh komen pengepala pada fail CPrototaip2Dlg.cpp adalah seperti di bawah:

```
/*
*****
Component Name : Prototaip2Dlg.cpp
Programmer : Mohd Hairul Nizam B. Md. Nasir
Contact : semangat_daulat@hotmail.com
Function : This component handles:
            1) Main interface of Plagiarizer
            2) Related to files:
               - doc.cpp(algorithm)
               - SettingDlg.cpp
               - FormDlg.cpp
            3) Send related value to CXdocument class
Version : 1.0 (2 February 2003)
*****
*/
```

2. Komen pada skrip aturcara

Selain itu juga, komen diletakkan pada setiap skrip aturcara bagi menerangkan struktur data yang terlibat serta kegunaan setiap pembolehubah pada setiap kelas. Contoh komen yang diletakkan dalam kelas CXdocument.cpp adalah seperti di bawah :

```

if( (ch>0x20) && (ch!=0x7F) )           // handle normal text characters
                                        // (including non-English)
{
    docbuffer[*docbuffercount]=ch;      // store the normal text character in
                                        // the buffer
    (*docbuffercount)++;                // increment the number of
                                        // characters in the buffer
}
else
    if( ch==LF )                        // handle a linefeed character (end
                                        // of paragraph)
    {
        docbuffer[*docbuffercount]=LF; // put the linefeed character
                                        // in the buffer
        (*docbuffercount)++;           // increment the number of
                                        // characters in the buffer
    }
}

```

3. Penamaan pembolehubah

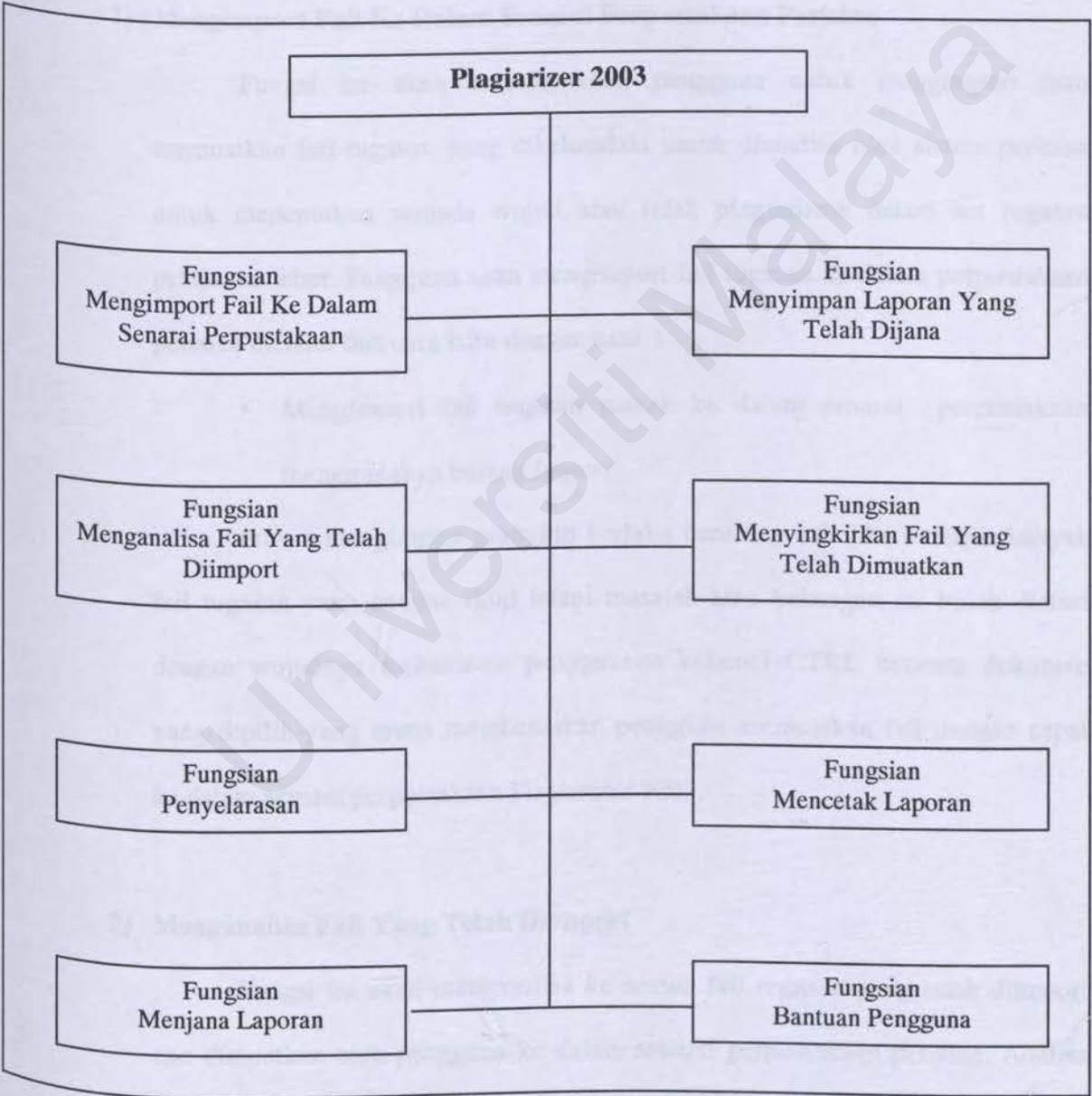
Setiap nama pembolehubah yang diberikan, kawalan dan modul dapat menyediakan identifikasi yang mudah kepada pengaturcara. Penamaan ini dilakukan dengan kod yang sejajar dan piawai.

3) Dokumentasi luaran

Dokumentasi luaran turut disediakan bagi setiap skrip aturcara terutamanya pada kelas yang penting seperti fail aturcara implementasi algoritma iaitu komponen kelas enjin pengesanan plagiarisme. Ini penting agar pengaturcaraan yang dilakukan dapat difahami dengan lebih terperinci.

6.4 Fasa Perlaksanaan Plagiarizer 2003

Setelah pengaturcaraan dilakukan, segala fungsi yang dibangunkan diintegrasikan dan dilaksanakan agar membentuk satu perisian yang sempurna. Segala fungsi yang dirancang dipastikan agar dilaksanakan dengan sempurna.



Rajah 6.3 : Pecahan Komponen Fungsian Bagi Plagiarizer 2003

Rajah 6.3 menunjukkan pecahan struktur fungsian bagi Plagiarizer 2003 yang mana setiap pecahan tersebut akan menerangkan keperluan atau kelakuan bagi sistem perisian tersebut. Secara umumnya, perisian Plagiarizer 2003 ini mempunyai 8 fungsian utama seperti yang ditunjukkan dalam rajah 6.3 iaitu :

1) Mengimport Fail Ke Dalam Senarai Perpustakaan Perisian

Fungsi ini akan membenarkan pengguna untuk mengimport atau memuatkan fail tugas yang dikehendaki untuk dianalisa oleh sistem perisian untuk menentukan samada wujud atau tidak plagiarisme dalam set tugas pelajar tersebut. Pengguna akan mengimport fail tugas ke dalam perpustakaan perisian melalui dua cara iaitu dengan cara :

- Mengimport fail tugas masuk ke dalam senarai perpustakaan menggunakan butang *Import*.

Proses mengimport mungkin berlaku berulang-kali jika terdapat banyak fail tugas yang hendak diuji tetapi masalah atau kekangan ini boleh diatasi dengan wujudnya mekanisme penggunaan kekunci CTRL berserta dokumen yang dipilih yang mana membenarkan pengguna memuatkan fail dengan cepat ke dalam senarai perpustakaan Plagiarizer 2003.

2) Menganalisa Fail Yang Telah Diimport

Fungsi ini akan menganalisa ke semua fail tugas yang telah diimport dan dimuatkan oleh pengguna ke dalam senarai perpustakaan perisian. Analisa dilakukan oleh sistem perisian untuk menentukan samada fail tugas yang dianalisa terlibat dengan plagiarisme atau tidak berdasarkan algoritma pengiraan

analisa yang dibangunkan. Analisa akan dibuat dengan mengklik pada butang *Analyze*.

3) Penyelarasan

Fungsi ini merupakan satu fungsi tambahan yang membenarkan pengguna menyelaras peraturan fungsi analisa sistem supaya tahap pengesanan plagiarisme adalah lebih tepat bergantung kepada kesesuaian keadaan tugas yang diberi. Penyelarasan peraturan yang boleh dibuat oleh pengguna ialah :

- 1) **Memilih samada untuk mengambilkira atau tidak, tanda baca dalam tugas semasa proses analisa.**

Tanda baca yang akan diambilkira oleh sistem merangkumi :

, . “ ‘ ; : ? / \ ! { } [] -

- 2) **Memilih samada untuk mengambilkira atau tidak, nombor dalam tugas semasa proses analisa**

Dalam contoh kes perkataan “10-bit” dan “8-bit” sistem akan menganggap perkataan tersebut adalah 2 perkataan yang sama jika pengguna tidak mengambilkira semasa proses analisa

4) Menyingkirkan fail yang telah dimuatkan

Fungsi ini membenarkan pengguna untuk menyingkirkan fail tugas yang telah dimuatkan ke dalam perpustakaan perisian. Penyingkiran fail boleh dilakukan melalui 2 cara iaitu :

- Penyingkiran ke semua senarai tugas yang terdapat di dalam perpustakaan perisian dengan mengklik pada butang *Clear All*
- Penyingkiran senarai tugas yang dipilih oleh pengguna sahaja dengan mengklik pada butang *Clear Selection*

5) Menjana laporan

Setelah proses analisa dilakukan oleh perisian ke atas senarai fail tugas yang dimuatkan ke dalam perpustakaan perisian, sistem perisian akan menjana laporan berasaskan HTML yang menunjukkan samada wujudnya plagiarisme atau tidak dalam set tugas yang telah diuji atau di analisa. Terdapat 3 jenis laporan yang akan dijana oleh perisian iaitu :

- Laporan keseluruhan yang menunjukkan peratusan plagiarisme yang terlibat antara set tugas yang diuji atau dianalisa dalam bentuk jadual
- Laporan berorientasikan graf
- Laporan terperinci antara 2 fail yang terlibat dengan plagiarisme.

6) Menyimpan laporan yang telah dijana

Pengguna mempunyai pilihan samaada untuk menyimpan atau tidak bukti laporan yang menunjukkan kes plagiarisme antara set tugas yang telah dianalisa yang mana tujuan penyimpanan laporan ini adalah untuk memudahkan pengguna untuk memanggil semula laporan tersebut tanpa perlu menguji semula

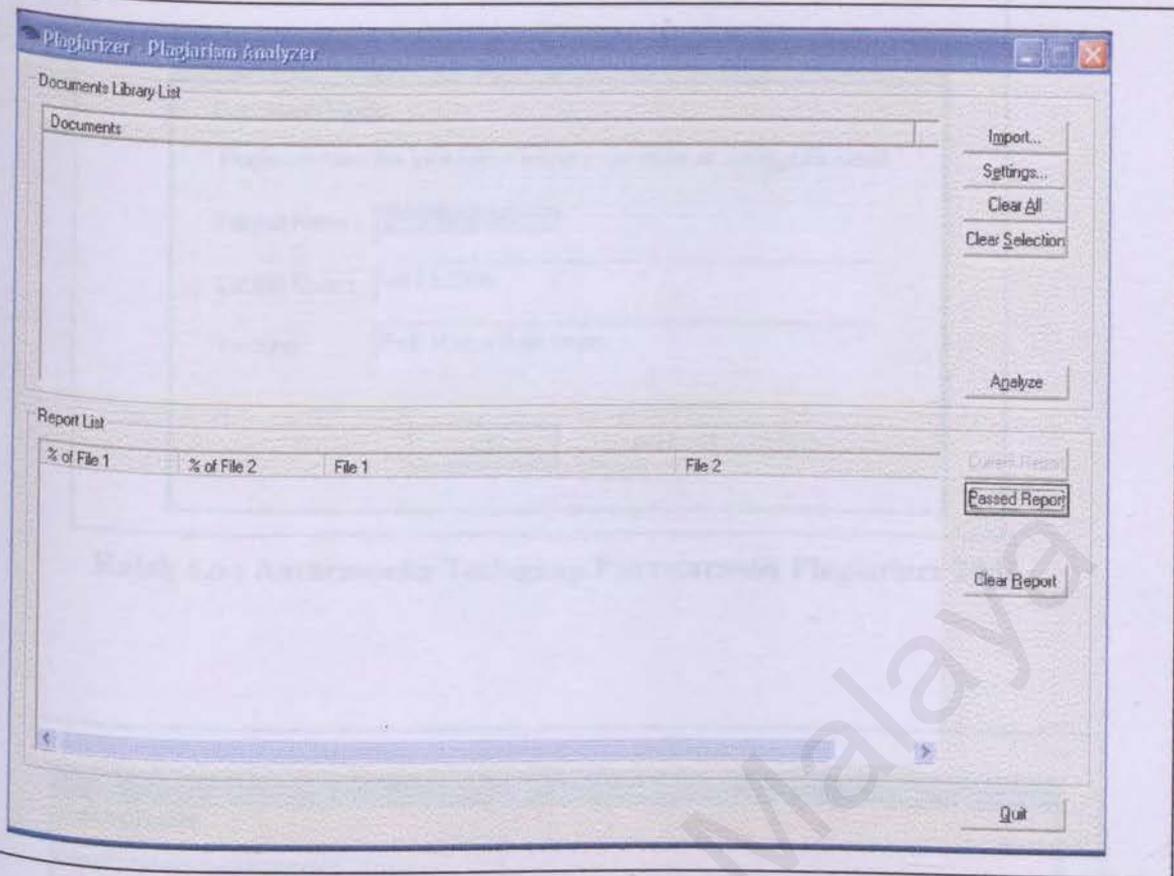
fail tugas tersebut. Penyimpanan ini boleh dilakukan dengan menggunakan fungsi simpan iaitu (*Save as*) yang terdapat pada pelayar web Internet Explorer.

7) Mencetak laporan

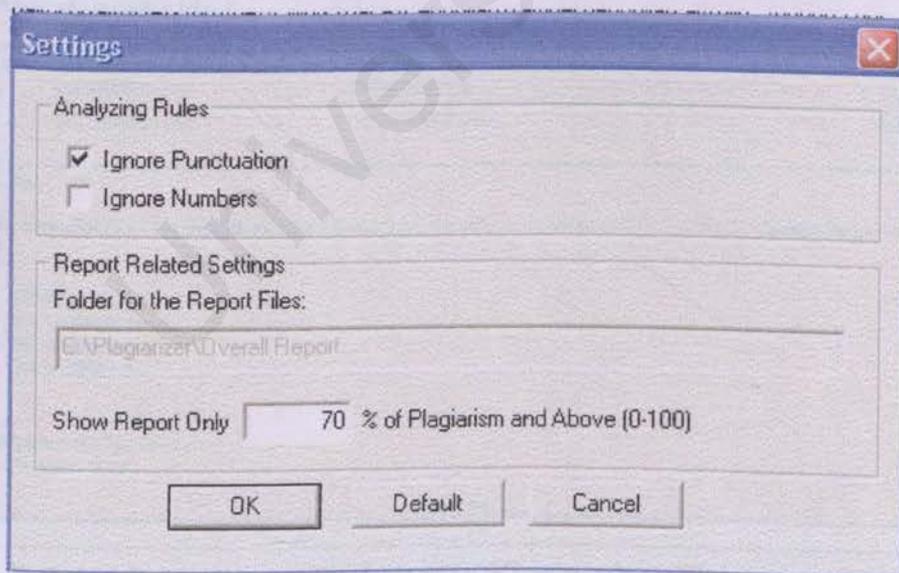
Fungsi ini membenarkan pengguna untuk mencetak laporan yang telah dijana oleh sistem bagi memudahkan pengguna meneliti laporan tersebut selain daripada boleh dijadikan bukti bahawa tugas tersebut terlibat dengan plagiarisme. Cetakan ini boleh dilakukan dengan menggunakan fungsi cetakan (*Print*) yang terdapat pada pelayar web Internet Explorer.

8) Bantuan pengguna

Fungsi bantuan pengguna bertujuan untuk memudahkan pengguna untuk mengetahui cara-cara untuk menggunakan sistem serta mengetahui apakah fungsi setiap elemen antaramuka yang ditunjukkan dalam Plagiarizer 2003.



Rajah 6.4 : Antaramuka Utama Plagiarizer 2003



Rajah 6.5 : Antaramuka Tetingkap Penyelarasan Plagiarizer 2003

Documents Profile Form

Documents Profile

Plagiarizer need the information below to generate an appropriate report

Subject Name :

Subject Code :

Lecturer :

Rajah 6.6 : Antaramuka Tetingkap Penyelarasan Plagiarizer 2003

Overall Report of Plagiarism

Analyzing Profiles

Subject : Komputeran Kembara(WRES3405)
 Lecturer : Dr. Mazliza
 Analyzing Date : Monday, 03-February-2003, 21:43:40
 Analyzed Documents : 6 documents
 Descriptions : Shows only 70% & above percentage of plagiarism
 Go to summary...

Percentage of Plagiarism Tables

File 1	File 2	% File 1	% File 2	Severity	Details
WES000001.doc	WES000002.doc	100.00 %	100.00 %	High	View
WES000005.doc	WES000006.doc	82.04 %	90.90 %	High	View

Summary

It is recommended to you to check more further about the 4 potential documents with high severity of plagiarism as stated below :

1. WES000001.doc
2. WES000002.doc
3. WES000005.doc
4. WES000006.doc

High severity: 2 possibilities
 Medium severity: 0 possibilities
 Show the graph report...

[Back to top]

Rajah 6.7 : Antaramuka Laporan Keseluruhan Plagiarizer 2003

Report of Plagiarism

Statistical Report

Graph shows only documents involved with 70% and above of plagiarism



Rajah 6.8 : Antaramuka Laporan Berorientasikan Graf Plagiarizer 2003

Report Details

Documents Statistik

File 1 : WES000001.doc

File 2 : WES000002.doc

Total Word Count File 1: 238 words

Total Word Count File 2: 238 words

Total matching : 238 words

Percentage matching file 1 against file 2: 100.00 %

Percentage matching file 2 against file 1: 100.00 %

Details

Matching Vocabulary Matching Frequency

April	4
B	1
Begitu	1
Dengan	4
Didapati	1
E	1
Feb	2
Jika	1
Julai	2
Kami	2
Kos	8
Maka	1
Penambahan	1

Rajah 6.9 : Antaramuka Laporan Terperinci Plagiarizer 2003

BAB 7

PENGUJIAN PERISIAN

PENGUJIAN PERISIAN

7.1 Pengenalan

Setelah selesai menulis aturcara, elemen yang paling penting untuk memastikan sama ada perisian yang baik perlu memenuhi kehendak pengguna dengan melalui proses pengujian. Perisian yang berkualiti mampu menjalani apa jua pengujian yang diberikan. Dengan adanya pengujian, spesifikasi-spesifikasi, rekabentuk dan pengkodan yang telah dilakukan sepanjang pembangunan sistem akan dapat dibuat penelitian semula. Ujian yang dijalankan akan dapat memastikan fungsi-fungsi yang dibina adalah bebas daripada sebarang masalah ralat supaya perisian akan dapat memberikan keputusan dan output yang baik dan tepat seperti yang dijangkakan. Sesuatu pengujian yang baik ialah ia dapat mengenalpasti ralat-ralat yang tidak dapat dikesan semasa fasa analisis, rekabentuk dan pengkodan. Aktiviti pengujian perlu dilakukan secara berterusan terhadap perisian mahupun sistem bagi memastikan produk akhir yang dihantar kepada pelanggan dan pengguna adalah suatu produk yang berkualiti, konsisten dan bebas daripada ralat.

- 3) Pengujian Perisian yang terdiri daripada pengujian kebolehgunaan, ketepatan dan prestasi
- 4) Pengujian pemasangan

7.2.1 Pengujian Unit

Oleh kerana Plagiarizer 2003 dibangunkan berdasarkan kepada pengaturcaraan berorientasikan objek, maka untuk pengujian unit, satu kelas dianggap sebagai satu unit. Matlamat pengujian unit adalah untuk memastikan setiap unit iaitu kelas tersebut beroperasi dalam menghasilkan output yang tepat dan betul berdasarkan input yang diberikan. . Jika pengujian peringkat ini dilakukan dengan teratur dan teliti, masalah-masalah lain pada peringkat pengujian yang seterusnya tidak akan berlaku.

Dalam pengaturcaraan Plagiarizer 2003, terdapat 6 kelas kesemuanya, tetapi 4 kelas atau unit sahaja yang dibuat pengujian iaitu :

- Kelas CPrototaip2Dlg
- Kelas CSettingDlg
- Kelas CFormDlg
- Kelas CXdocument

Di mana keempat-empat unit atau kelas ini dicipta atau dijana oleh saya sendiri yang mana ia sudah tentu memerlukan pengujian sementara 2 kelas lagi dicipta secara automatik oleh Microsoft Visual C++ 6.0 yang mana tidak memerlukan pengujian.

Dalam pengujian unit ini, antara aktiviti pengujian yang telah dilakukan ialah:

1. Pengujian kod - Ujian ini dilakukan melalui pembacaan dan pengamatan semula kod yang telah di tulis bagi mengesan kesalahan sintaks yang berlaku.
2. Larian kod - Kod aturcara akan dikompil dan sekiranya terdapat ralat di dalam aplikasi tersebut, mesej akan dipaparkan . Ini untuk memastikan semua ralat sintaks dihapuskan.
3. Pengujian nilai pembolehubah – Ini dilakukan menggunakan fungsi nyahpijat (*debug*) yang tersedia pada Microsoft Visual C++. Ia bertujuan agar nilai pembolehubah bagi setiap kelas atau unit adalah tepat sepanjang masa pelaksanaan program. Nilai setiap pembolehubah yang terlibat akan dipantau sepanjang pelaksanaan program sehingga program itu ditamatkan.
4. Pengujian peruntukan memori – Setiap pembolehubah yang terlibat dengan peruntukan memori secara dinamik dipantau dan diuji agar setiap peruntukan memori yang telah dibuat dilepaskan setiap kali program tersebut tamat. Ini bertujuan untuk mengelak masalah kebocoran memori(*leaking memory*) dalam perisian Plagiarizer 2003 yang mana boleh menjejaskan kualiti akhir produk

Di antara kesilapan yang boleh dikesan dalam pengujian unit :

- Kesalahan dalam pengawalan logik
- Kesalahan sintaks
- Kesalahan pengiraan algoritma

7.2.2 Pengujian Integrasi

Dalam pengujian integrasi ke semua kelas atau unit yang telah diuji diintegrasikan untuk membentuk satu sistem perisian yang boleh bekerja dalam menghasilkan output akhir yang dikehendaki.

Dalam pengujian integrasi Plagiarizer 2003, apa yang dititikberatkan ialah agar setiap kelas dapat menghantar nilai kepada kelas lain tanpa wujudnya :

- Kehilangan nilai data semasa penghantaran nilai
- Kesilapan nilai data semasa penghantaran nilai

Pendekatan yang digunakan untuk melakukan pengujian integrasi ialah pendekatan *big-bang* yang mana setiap unit yang telah diuji secara berasingan dan akhir sekali setiap unit dicantumkan serentak membentuk satu modul sistem perisian yang besar.

7.2.3 Pengujian Perisian

Setelah pengujian unit dan integrasi telah dilakukan, maka aktiviti pengujian seterusnya ialah pengujian perisian. Oleh kerana matlamat dan rekabentuk akhir perisian Plagiarizer 2003 didasarkan kepada 3 konsep iaitu :

- Kebolehgunaan yang tinggi
- Ketepatan laporan yang tinggi
- Prestasi dan masa tindakbalas yang tinggi

Maka untuk pengujian perisian, ketiga-tiga perkara ini telah difokuskan agar produk akhir merupakan suatu produk yang mempunyai ciri-ciri di atas.

7.2.3.1 Pengujian Kebolehgunaan

Pengujian kebolehgunaan adalah bertujuan untuk menilai sejauh manakah perisian Plagiarizer 2003 mudah digunakan oleh pengguna. Ini dilakukan agar produk perisian akhir yang dihantar kepada pengguna mempunyai kualiti kebolehgunaan yang tinggi dengan membekalkan antaramuka yang mudah dan sepatutnya. Matlamat utama pengujian kebolehgunaan di sini adalah untuk mengetahui '*learning curve*' seterusnya memperbaiki antaramuka perisian Plagiarizer 2003 termasuk laporan yang dijana oleh perisian berdasarkan komen dan pendapat pengguna supaya tahap pembelajaran menggunakan perisian ini adalah singkat.

7.2.3.1.1 Asas Persekitaran Pengujian Kebolehgunaan

Untuk pengujian kebolehgunaan ini, 5 orang pengguna telah dipilih yang terdiri daripada tutor Fakulti Sains Komputer & Teknologi Maklumat, Universiti Malaya di mana ukuran kebolehgunaan sistem ini adalah unit masa di mana berapa lama masa yang diambil oleh pengguna untuk mempelajari menggunakan perisian Plagiarizer 2003 ini.

Atas tujuan tersebut, pengujian dibahagikan kepada 2 iaitu:

- 1) **Kes Pengujian 1** : Berapa lama masa diambil oleh pengguna untuk mahir dengan fungsi-fungsi asas Plagiarizer 2003
- 2) **Kes Pengujian 2** : Berapa lama masa diambil oleh pengguna untuk mahir dengan fungsi-fungsi tambahan(*advanced*) Plagiarizer 2003

Fungsi-fungsi asas Plagiarizer 2003 merangkumi :

- Bagaimana menggunakan Plagiarizer 2003 untuk mengesan plagiarisme merangkumi fungsi setiap butang asas seperti butang *Import, Analyze, Clear All, Clear Selection, Clear Report, Current Report, Passed Report, Settings*.
- Kegunaan dan fungsi setiap elemen dalam antaramuka
- Melihat dan membaca kesemua laporan yang dijana oleh Plagiarizer 2003

7.2.1.1.2 Hasil Keputusian Percobaan Kecepatan

Fungsi-fungsi tambahan (*Advanced*) Plagiarizer 2003 merangkumi :

- Penggunaan kekunci pantas (*Acceleration Key*) dan kekunci *Tab* selain menggunakan tetikus untuk melakukan atau membuat suatu perkara atau aksi
- Membuat konfigurasi aturan yang terdapat pada tettingkap penyelarasan (*Settings*) bagi mendapatkan laporan yang lebih tepat dan menyeluruh bersesuaian keadaan dengan dokumen yang hendak diuji
- Teknik melihat dan membaca kesemua laporan yang dijana secara efisien berdasarkan kepada setiap seksyen yang terdapat dalam laporan tersebut.

No	Penggunaan	7.2.1.1.2	4.0 minit
1	Penggunaan	7.2.1.1.2	4.0 minit
2	Penggunaan	7.2.1.1.2	4.3 minit
3	Penggunaan	7.2.1.1.2	4.3 minit
4	Penggunaan	7.2.1.1.2	4.3 minit
5	Penggunaan	7.2.1.1.2	4.3 minit
	Purata	7.2.1.1.2	4.30 minit

Jadual 7.1 : Masa yang diambil untuk Maklup Dokumen Dengan Fungsi

Plagiarizer 2003

"Masa diambil tanpa bantuan" Masa yang diambil oleh pengguna untuk mempelajari dan menguasai fungsi-fungsi asas Plagiarizer tanpa bantuan daripada pengguna.

"Masa diambil dengan bantuan" Masa yang diambil oleh pengguna untuk mempelajari dan menguasai fungsi-fungsi asas Plagiarizer dengan bantuan daripada pengguna oleh pengguna.

7.2.3.1.2 Hasil Keputusan Pengujian Kebolegunaan

Keputusan Pengujian 1:

Berapa Lama Masa Diambil Oleh Pengguna Untuk Mahir Dengan Fungsi-Fungsi Asas

Plagiarizer 2003

No.	Pengguna	*Masa Diambil	
		Tanpa Bimbingan	Dengan Bimbingan
1.	Pengguna Pertama	7.5 minit	4.0 minit
2.	Pengguna Kedua	8.0 minit	4.5 minit
3.	Pengguna Ketiga	10.0 minit	6.0 minit
4.	Pengguna Keempat	8.5 minit	4.5 minit
5.	Pengguna Kelima	7.0 minit	3.5 minit
	Purata	8.20 minit	4.50 minit

Jadual 7.1 : Masa Yang Diambil Untuk Mahir Dengan Fungsi Asas

Plagiarizer 2003

*Masa diambil tanpa bimbingan : Masa yang diambil oleh pengguna untuk mempelajari dan menguasai fungsi-fungsi asas Plagiarizer tanpa sebarang bantuan pengajar.

*Masa diambil dengan bimbingan : Masa yang diambil oleh pengguna untuk mempelajari dan menguasai fungsi-fungsi asas Plagiarizer dengan bantuan dan penjelasan oleh pengajar.

Keputusan Pengujian 2:

Berapa Lama Masa Diambil Untuk Pengguna Mahir Dengan Fungsi-Fungsi

Tambahan(*Advanced*) Plagiarizer 2003

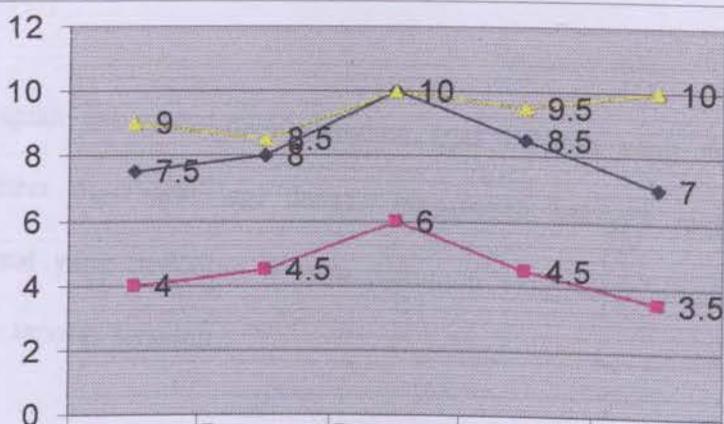
No.	Pengguna	Masa Diambil Dengan Bimbingan
1.	Pengguna Pertama	9.0 minit
2.	Pengguna Kedua	8.5 minit
3.	Pengguna Ketiga	10.0 minit
4.	Pengguna Keempat	9.5 minit
5.	Pengguna Kelima	9.0 minit
	Purata	9.2 minit

Jadual 7.2 : Masa Yang Diambil Untuk Mahir Dengan Fungsi

Tambahan(*Advanced*) Plagiarizer 2003

*Masa diambil dengan bimbingan : Masa yang diambil oleh pengguna untuk mempelajari dan menguasai fungsi-fungsi tambahan(*Advanced*) Plagiarizer dengan bantuan dan penjelasan oleh pengajar.

Masa Pembelajaran, (minit)



	Pengguna 1	Pengguna 2	Pengguna 3	Pengguna 4	Pengguna 5
—●— Masa Pembelajaran Fungsi Biasa Tanpa Bimbingan	7.5	8	10	8.5	7
—■— Masa Pembelajaran Fungsi Biasa Dengan Bimbingan	4	4.5	6	4.5	3.5
—▲— Masa Pembelajaran Fungsi Tambahan Dengan Bimbingan	9	8.5	10	9.5	10

Rajah 7.2 : Graf Masa Pembelajaran Bagi Kedua-dua Kes Pengujian

Lain-Lain Keputusan

Sepanjang menjalankan kes pengujian bersama kelima-lima orang pengguna tersebut, didapati bahawa:

- 1) Kesemua pengguna memberi komen bahawa perisian ini membekalkan antaramuka yang mudah untuk digunakan di samping fungsi yang dibekalkan adalah cukup untuk membenarkan pengesanan plagiarisme tetapi manual pengguna harus dibekalkan bersama.

- 2) Sebahagian pengguna mencadangkan agar laporan yang dijana oleh Plagiarizer diperbaiki lagi dengan menambah seksyen rumusan dan maklumat yang berkaitan dengan dokumen yang telah dianalisa serta struktur laporan tersebut.

7.2.3.1.3 Kesimpulan Pengujian Kebolehgunaan

Berdasarkan pengujian di atas di dapati bahawa pengguna mengambil masa yang singkat untuk menguasai Plagiarizer 2003 dan ini menunjukkan bahawa Plagiarizer 2003 mudah untuk digunakan samada wujudnya tenaga pengajar untuk menjelaskan setiap fungsi asas mahupun tidak.

Selain itu, komen dan pendapat oleh pengguna iaitu tutor-tutor seperti yang telah dinyatakan di atas telah diambil kira dan tindakan untuk memperbaiki Plagiarizer 2003 telah dilakukan.

7.2.3.2 Pengujian Ketepatan Laporan

Matlamat pengujian ketepatan laporan perisian Plagiarizer 2003 adalah untuk menentukan bahawa perisian Plagiarizer 2003 ini berfungsi dengan tepat dari segi penjana laporan dan maklumat oleh perisian Plagiarizer 2003 itu sendiri berdasarkan kepada model algoritma yang telah dibangunkan dan telah ditetapkan dahulu. Jika sistem perisian ini dapat menjana dan menentukan peratusan plagiarisme sama seperti pengiraan model algoritma tersebut, perisian ini dikatakan telah berfungsi dengan tepat dan seperti yang dikehendaki dalam menentukan tahap peratusan plagiarisme.

7.2.3.2.1 Asas Persekitaran Pengujian Ketepatan Laporan

Untuk tujuan tersebut, 2 komponen utama laporan telah diuji dalam menentukan ketepatan operasi perisian Plagiarizer 2003 ini iaitu:

- 1) **Kes Pengujian 1** : Pengujian dalam menentusahkan ketepatan bilangan laporan yang dijana oleh perisian
- 2) **Kes Pengujian 2** : Pengujian dalam menentusahkan ketepatan laporan peratusan berdasarkan jenis aturan analisa yang dijana oleh perisian

Kedua-dua kes pengujian ini, diuji berdasarkan kepada spesifikasi kes pengujian seperti dalam jadual bawah iaitu :

Perkara	Nilai	Komen
Sumber sampel pengujian	Tugasan Pelajar (subjek : Pengurusan Projek Sesi 2001/2002)	Sumber diambil dari Prof. Madya Raja Ainon
Bilangan sampel pengujian	14 sampel	Dilabelkan dari : sampel A hingga N
Sampel Dasar Perbandingan	Sampel A	Sampel A menjadi asas kepada semua perbandingan sampel

Jadual 7.3 : Spesifikasi Kes Pengujian Ketepatan Laporan Plagiarizer 2003

Ketiga-tiga bahagian ini akan dibuktikan kes pengujian masing-masing berdasarkan kepada model algoritma yang telah dibangunkan dahulu di mana pengujian didasarkan kepada Sampel A. Keseluruhan konsep dan pengiraan algoritma dilampirkan dalam bahagian lampiran.

7.2.3.2.2 Aktiviti Pengujian, Keputusan, Pembuktian Dan Kesimpulan

Kes Pengujian 1 :

Pengujian Dalam Menentukan Ketepatan Bilangan Laporan Yang Dijana Oleh Perisian Plagiarizer 2003

Kes pengujian ini adalah untuk menentukan bahawa sistem menjana bilangan laporan yang sepatutnya berdasarkan kepada input iaitu bilangan sampel yang diuji.

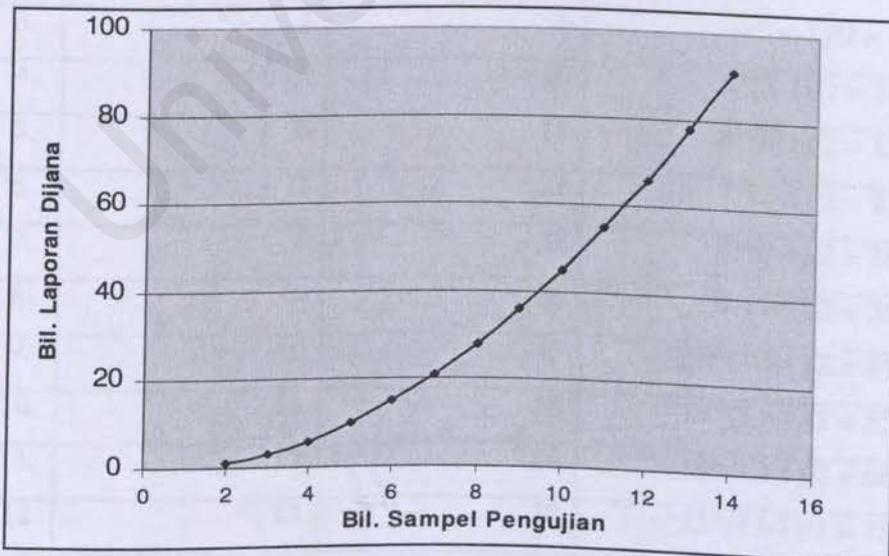
Kaedah:

- 1) Kesemua sampel diuji di mana 2 sampel pertama diuji terlebih dahulu iaitu sampel A dan Sampel B. Bilangan laporan yang dijana oleh perisian direkodkan.
- 2) Setelah 2 sampel tersebut diuji, pengujian seterusnya pula melibatkan bilangan sampel sebanyak 3 pula dan segala keputusan direkodkan sehingga keempat belas-empat belas sampel tersebut menjalani pengujian yang sama.
- 3) Hasil keputusan iaitu bilangan laporan yang dijana bagi setiap bilangan sampel yang diuji direkod dan dijadualkan.

Hasil Keputusan Pengujian Kes 1:

No.	Bilangan Sampel Pengujian	Bilangan Laporan yang Dijana
1.	2	1
2.	3	3
3.	4	6
4.	5	10
5.	6	15
6.	7	21
7.	8	28
8.	9	36
9.	10	45
10.	11	55
11.	12	66
12.	13	78
13.	14	91

Jadual 7.4 : Jumlah Bilangan Laporan yang Dijana oleh Sistem Perisian Berdasarkan Bilangan Sampel



Rajah 7.3 : Graf Bilangan Laporan Dijana Lawan Bilangan Sampel Pengujian

Analisa Pembuktian Hasil Keputusan Kes Pengujian 1:

Untuk membuktikan hasil keputusan yang dijana adalah sama dengan model algoritma yang dibangunkan, kaedah pengiraan manual akan digunakan atas tujuan tersebut yang mana hasil melalui kaedah pengiraan manual akan dibandingkan dengan hasil keputusan sebenar yang dijana oleh perisian Plagiarizer 2003. Model algoritma yang digunakan untuk tujuan pembuktian kes ialah :

$$\text{Jumlah Bilangan Laporan yang dijana} = [n(n-1)]/2$$

n=bilangan sampel untuk di analisa

Melalui kaedah pengiraan manual:

No.	Bil. Sampel Pengujian	Bil. Laporan yang Dijana	Bil. Laporan Sepatutnya
1.	2	1	$2*(2-1)/2 = 1$
2.	3	3	$3*(3-1)/2 = 3$
3.	4	6	$4*(4-1)/2 = 6$
4.	5	10	$5*(5-1)/2 = 10$
5.	6	15	$6*(6-1)/2 = 15$
6.	7	21	$7*(7-1)/2 = 21$
7.	8	28	$8*(8-1)/2 = 28$
8.	9	36	$9*(9-1)/2 = 36$
9.	10	45	$10*(10-1)/2 = 45$
10.	11	55	$11*(11-1)/2 = 55$
11.	12	66	$12*(12-1)/2 = 66$
12.	13	78	$13*(13-1)/2 = 78$
13.	14	91	$14*(14-1)/2 = 91$

Jadual 7.5 : Hasil Keputusan Melalui Pengiraan Manual

Kesimpulan Kes Pengujian 1

Melalui perbandingan hasil keputusan antara hasil keputusan sebenar yang dijana oleh sistem perisian Plagiarizer 2003 dengan hasil keputusan melalui pengiraan sebenar didapati bahawa hasil keputusan adalah sama. Maka, ini membuktikan bahawa perisian Plagiarizer 2003 menjana bilangan laporan dengan tepat.

Kes Pengujian 2:

Pengujian Dalam Menentusahkan Ketepatan Laporan Berdasarkan Aturan Analisa Yang Dijana Oleh Perisian

Pengujian dalam bahagian ini adalah untuk menentusahkan bahawa sistem menjana laporan peratusan plagiarisme dengan tepat berdasarkan 4 peraturan analisa berikut iaitu:

- **Aturan A** : Analisa dengan faktor mengabaikan tanda baca
- **Aturan B** : Analisa dengan faktor mengabaikan nombor
- **Aturan C** : Analisa dengan faktor mengabaikan tanda baca dan nombor
- **Aturan D** : Analisa tanpa sebarang faktor pengabaian.

Kaedah :

- 1) Kesemua sampel dianalisa secara serentak berdasarkan 4 jenis pengujian seperti yang dinyatakan di atas.
- 2) Setiap jenis pengujian yang tersebut, laporan mengenai jumlah bilangan perkataan, jumlah bilangan perkataan yang sama dan peratusan plagiarisme direkodkan.
- 3) Kesemua hasil keputusan dijadualkan.

Hasil Keputusan Pengujian Kes 2:

Aturan A: Analisa Dengan Faktor Mengabaikan Tanda Baca

No.	Sampel	Jumlah Perkataan	Jumlah Perkataan Sama	Peratusan %
1.	Sampel B	304	304	100.00
2.	Sampel C	208	99	47.60
3.	Sampel D	126	60	47.62
4.	Sampel E	268	105	39.18
5.	Sampel F	244	102	41.80
6.	Sampel G	82	42	51.22
7.	Sampel H	304	304	100.00
8.	Sampel I	204	83	40.69
9.	Sampel J	157	61	38.85
10.	Sampel K	265	85	32.08
11.	Sampel L	233	107	45.92
12.	Sampel M	253	110	83.00
13.	Sampel N	286	225	78.67

Jadual 7.6 : Hasil Keputusan Kes Pengujian 2 Bagi Aturan A

Aturan B: Analisa Dengan Faktor Mengabaikan Nombor

No.	Sampel	Jumlah Perkataan	Jumlah Perkataan Sama	Peratusan %
1.	Sampel B	264	264	100.00
2.	Sampel C	186	73	39.25
3.	Sampel D	118	51	43.22
4.	Sampel E	250	77	30.80
5.	Sampel F	224	80	35.71
6.	Sampel G	77	30	38.96
7.	Sampel H	264	264	100.00
8.	Sampel I	187	58	31.02
9.	Sampel J	150	34	22.67
10.	Sampel K	253	65	25.69
11.	Sampel L	210	80	38.10
12.	Sampel M	221	173	78.28
13.	Sampel N	252	184	73.02

Jadual 7.7 : Hasil Keputusan Kes Pengujian 2 Bagi Aturan B

Aturan C: Analisa Dengan Faktor Mengabaikan Tanda Baca dan Nombor

No.	Sampel	Jumlah Perkataan	Jumlah Perkataan Sama	Peratusan %
1.	Sampel B	238	238	100.00
2.	Sampel C	173	70	40.46
3.	Sampel D	107	48	44.86
4.	Sampel E	244	92	37.70
5.	Sampel F	220	90	40.91
6.	Sampel G	68	32	47.06
7.	Sampel H	238	238	100.00
8.	Sampel I	175	64	36.57
9.	Sampel J	139	51	36.69
10.	Sampel K	252	81	32.14
11.	Sampel L	210	89	42.38
12.	Sampel M	202	162	80.20
13.	Sampel N	232	174	75.00

Jadual 7.8 : Hasil Keputusan Kes Pengujian 2 Bagi Aturan C

Aturan D: Analisa Tanpa Sebarang Faktor Pengabaian

No.	Sampel	Jumlah Perkataan	Jumlah Perkataan Sama	Peratusan %
1.	Sampel B	278	278	100.00
2.	Sampel C	188	71	37.77
3.	Sampel D	119	48	40.34
4.	Sampel E	256	78	30.47
5.	Sampel F	236	83	35.17
6.	Sampel G	80	28	35.00
7.	Sampel H	278	278	100.00
8.	Sampel I	193	58	30.05
9.	Sampel J	151	30	19.87
10.	Sampel K	257	69	26.85
11.	Sampel L	217	83	38.25
12.	Sampel M	235	181	77.02
13.	Sampel N	267	194	72.66

Jadual 7.9 : Hasil Keputusan Kes Pengujian 2 Bagi Aturan D

Analisa Pembuktian Hasil Keputusan Kes Pengujian 2:

Berdasarkan kepada keempat-empat hasil keputusan yang telah dijadualkan, maka laporan yang dijana oleh perisian Plagiarizer akan dibuktikan di mana kaedah pengiraan manual akan digunakan untuk mengira **bilangan perkataan** bagi setiap sampel dan **bilangan perkataan yang sama** antara sampel yang diuji dengan sampel A.

Kaedah :

- 1) Melalui kaedah pengiraan manual, bilangan perkataan bagi setiap sampel dan bilangan perkataan atau frekuensi yang perkataan yang sama antara sampel yang diuji dengan sampel A ditentukan.
- 2) Berdasarkan kepada kedua-dua data ini, peratusan plagiarisme ditentukan melalui pengiraan manual berdasarkan formula di bawah dan hasil pengiraan dijadualkan seperti yang di bawah:
- 3) Kaedah ini diulang dari bagi set **Aturan B** hingga **Aturan D**

$$\text{Peratusan plagiarisme bagi tugas, \%} = \frac{\text{Bilangan Perkataan yang Sama antara 2 Sampel}}{\sum \text{Bilangan Perkataan dalam Sampel}}$$

No.	Sampel	Peratusan yang Dijana oleh sistem %	Peratusan Melalui pengiraan manual %
1.	Sampel B	100.00	$(238/238)*100 = 100.00$
2.	Sampel C	40.46	$(70/173)*100 = 40.46$
3.	Sampel D	44.86	$(48/107)*100 = 44.86$
4.	Sampel E	37.70	$(92/244)*100 = 37.70$
5.	Sampel F	40.91	$(90/220)*100 = 40.91$
6.	Sampel G	47.06	$(32/68)*100 = 47.06$
7.	Sampel H	100.00	$(238/238)*100 = 100.00$
8.	Sampel I	36.57	$(64/175)*100 = 36.57$
9.	Sampel J	36.69	$(51/139)*100 = 36.69$
10.	Sampel K	32.14	$(81/252)*100 = 32.14$
11.	Sampel L	42.38	$(89/210)*100 = 42.38$
12.	Sampel M	80.20	$(162/202)*100 = 80.20$
13.	Sampel N	75.00	$(174/232)*100 = 75.00$

Jadual 7.10 : Perbandingan Hasil Laporan Yang Dijana Dengan

Hasil Pengiraan Manual Untuk Aturan A

Kesimpulan Hasil Kes Pengujian 2

Melalui pengiraan manual yang telah dibuat, didapati bahawa laporan pengiraan peratusan plagiarisme yang dijana oleh perisian Plagiarizer 2003 adalah sama dengan peratusan plagiarisme melalui pengiraan manual berdasarkan model algoritma yang telah dibangunkan. Maka ini membuktikan bahawa laporan peratusan yang dijana oleh perisian Plagiarizer 2003 adalah tepat.

7.2.3.3 Pengujian Prestasi

Pengujian prestasi adalah bertujuan untuk menilai prestasi perisian Plagiarizer 2003 dalam membuat proses analisa dalam menentukan tahap plagiarisme dalam set sampel tugas yang diuji. Ini dilakukan agar produk perisian akhir yang dihantar kepada pelanggan mempunyai kualiti prestasi yang tinggi selain daripada membekalkan antaramuka yang mudah serta ketepatan laporan yang tinggi.

Matlamat pengujian prestasi yang dilakukan difokuskan kepada 2 perkara penting iaitu untuk:

- 1) Memastikan agar Plagiarizer 2003 memulangkan masa yang munasabah dalam melakukan proses analisa berdasarkan kepada spesifikasi komputer pengguna
- 2) Memastikan agar Plagiarizer 2003 tidak mendatangkan masalah jika pengguna menggunakan versi Windows yang berlainan iaitu Windows 98, 98SE, 2000 dan XP

7.2.3.3.1 Asas Persekitaran Pengujian Prestasi

Prestasi perisian Plagiariizer 2003 diuji dalam unit masa pemproses di mana masa tindakbalas merupakan masa yang diambil dari satu titik di mana sistem perisian mula memuat dan membaca dokumen sehingga ia siap menjana laporan peratusan plagiarisme di mana aturan analisa mengambil nilai *default* perisian.

Untuk tujuan tersebut, kes pengujian akan dilakukan ke atas 4 komputer yang berlainan kemampuannya dan spesifikasinya untuk mengetahui tahap prestasi analisa perisian. Ini bertujuan untuk menentukan bahawa perisian ini memulangkan nilai prestasi yang munasabah berdasarkan kepada spesifikasi komputer pengguna. Spesifikasi komputer yang digunakan dalam persekitaran pengujian adalah seperti dalam jadual yang di bawah :

Komputer	Kelajuan Pemproses	Memori	Platform
A	AMD Duron (800Mhz)	384.00 SDRAM	Windows 2000
B	Pentium III(550Mhz)	256.00 SDRAM	Windows ME
C	Pentium (200Mhz)	32.00 SDRAM	Windows 98SE
D	Pentium IV(2.2Ghz)	128.00 DDRAM	Windows XP

Jadual 7.11 : Spesifikasi Komputer Yang Digunakan Dalam

Persekitaran Pengujian

Masa diukur dan direkodkan menggunakan perpustakaan bahasa C di bawah fail pengepala *time.h* yang berkebolehan merekod sela masa Unit Pemprosesan Pusat.

Untuk memastikan proses pengujian dilakukan secara menyeluruh, kes pengujian akan dibahagikan kepada 3 kes yang mana masing-masing diklasifikasikan mengikut tahap bebanan masing-masing iaitu tahap bebanan rendah, sederhana dan tinggi. Perisian Plagiarizer 2003 akan membuat analisa ke atas ketiga-tiga jenis pengujian tersebut yang mana keputusan akan direkodkan dan dijadualkan.

Berikut adalah 3 jenis pengujian iaitu :

- 1) **Kes Pengujian 1** : Tahap bebanan rendah berdasarkan spesifikasi jadual di bawah:

Perkara	Nilai	Komen
Sumber Sampel	Tugasan Pelajar (subjek : Pengurusan Projek Sesi 2001/2002)	Sumber diambil dari Prof. Madya Raja Ainon
Bilangan karakter	Dari 1,900 sehingga 2,200 karakter	
Bilangan perkataan	Dari 280 sehingga 350 patah perkataan	
Bilangan mukasurat	1 mukasurat	
Bilangan sampel	15 sampel	

Jadual 7.12 : Persekitaran Kes Pengujian 1

(Tahap Bebanan Rendah)

2) **Kes Pengujian 2** : Tahap bebanan sederhana berdasarkan spesifikasi jadual di bawah:

Perkara	Nilai	Komen
Sumber Sampel	Tugasan Pelajar (subjek : Termodinamik Sesi 2002/2003)	Sumber diambil dari 10 pelajar Fakulti Kejuruteraan, Jabatan Kimia
Bilangan Karakter	Dari 11,000 hingga 13,500 karakter	
Bilangan Perkataan	Dari 1,800 sehingga 2,400 patah perkataan	
Bilangan Mukasurat	7 mukasurat	
Bilangan Sampel	30 sampel (Secara iteratif 10 sampel sebenar didarab 3)	

Jadual 7.13 : Persekitaran Kes Pengujian 2

(Tahap Bebanan Sederhana)

3) **Kes Pengujian 3** : Tahap bebanan tinggi berdasarkan spesifikasi jadual di bawah:

Perkara	Nilai	Komen
Sumber Sampel	Tugasan Pelajar (subjek : Keselamatan Komputer Sesi 2002/2003)	Sumber diambil dari 25 pelajar, Fakulti Sains Komputer
Bilangan Karakter	Dari 30,000 hingga 35,000 karakter	
Bilangan Perkataan	Dari 6,000 sehingga 8,000 patah perkataan	
Bilangan Mukasurat	20 mukasurat	
Bilangan Sampel	100 sampel (Secara iteratif 25 sampel sebenar didarab 4)	

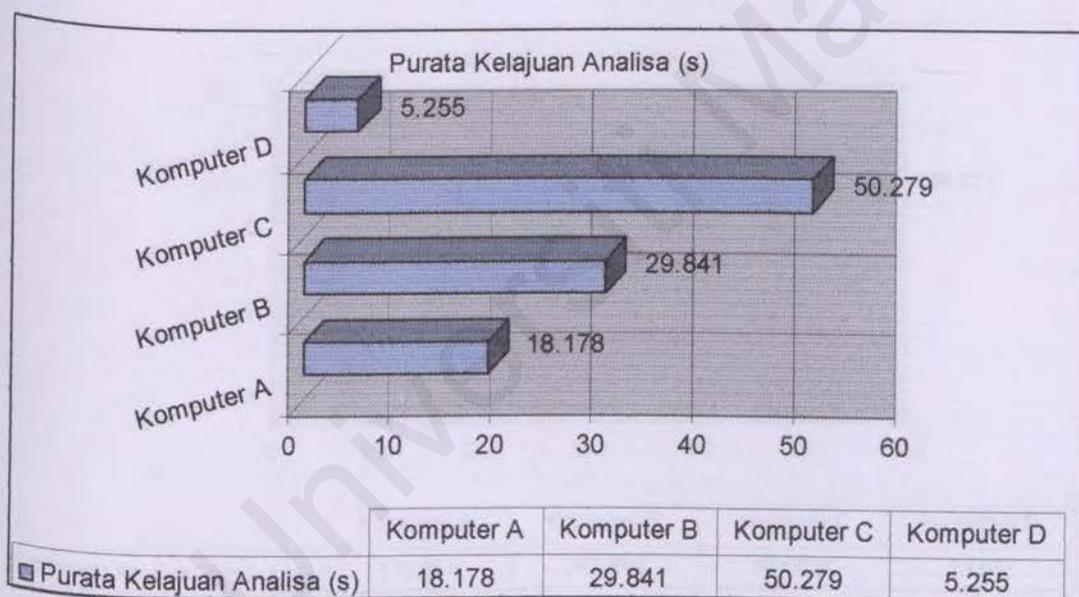
Jadual 7.14 : Persekitaran Kes Pengujian 3

(Tahap Bebanan Tinggi)

Keputusan Pengujian Kes 2 (Tahap Bebanan Sederhana):

Komputer	Masa T_1 (s)	Masa T_2 (s)	Masa T_3 (s)	Masa T_4 (s)	Masa T_5 (s)	Masa T_6 (s)	Purata T_{Avg} (s)
A	18.236	18.112	18.198	18.21	18.112	18.2	18.178
B	30.045	29.997	29.871	30.121	29.874	29.14	29.841
C	50.33	49.221	50.911	51.989	50.003	49.22	50.279
D	5.112	5.112	5.719	4.998	5.322	5.267	5.255

Jadual 7.16 : Hasil Keputusan Kes Pengujian 2 (Tahap Bebanan Sederhana)

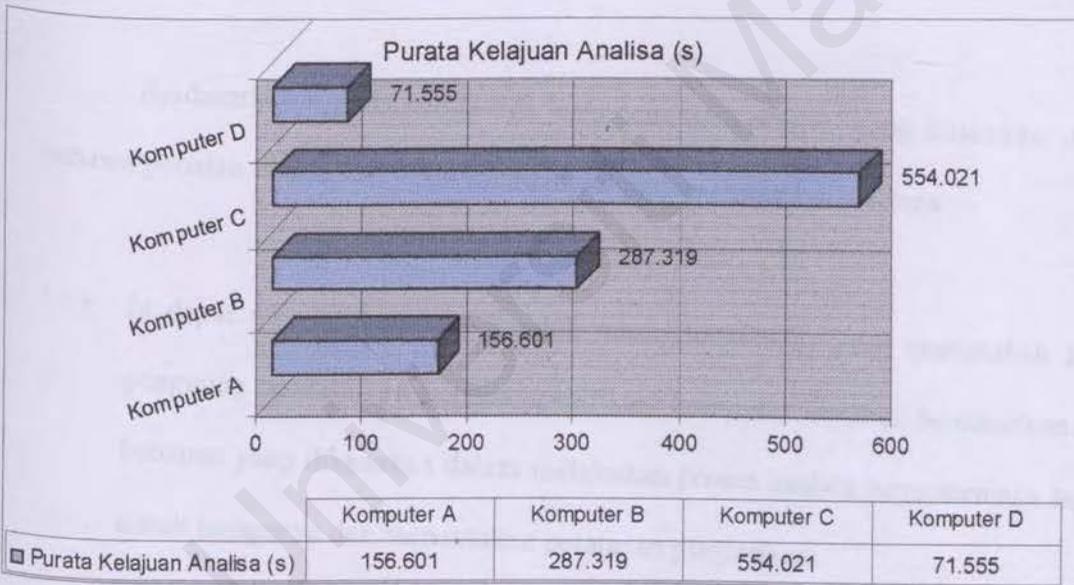


**Rajah 7.5 :Graf Purata Kelajuan Masa Kes Pengujian 2
(Tahap Bebanan Sederhana)**

Keputusan Pengujian Kes 3 (Tahap Bebanan Tinggi):

Komputer	Masa T_1 (s)	Masa T_2 (s)	Masa T_3 (s)	Masa T_4 (s)	Masa T_5 (s)	Masa T_6 (s)	Purata T_{Avg} (s)
A	155.237	157.236	156.998	157.012	157.112	156.011	156.601
B	287.882	286.771	288.012	286.559	286.921	287.771	287.319
C	551.211	553.989	555.912	554.989	553.011	555.012	554.021
D	71.532	72.321	70.411	71.333	71.414	72.321	71.555

Jadual 7.17 : Hasil Keputusan Kes Pengujian 3 (Tahap Bebanan Tinggi)



**Rajah 7.6 :Graf Purata Kelajuan Masa Kes Pengujian 3
(Tahap Bebanan Tinggi)**

Lain-lain Keputusan

Sepanjang melakukan ketiga-tiga kes pengujian, didapati bahawa tiada sebarang masalah keserasian antara perisian Plagiarizer 2003 dengan platform sistem pengendalian Windows yang berlainan di mana pengujian yang dibuat ke atas komputer yang berplatformkan Windows 98SE, Windows ME, Windows 2000 dan Windows XP walaupun Plagiarizer 2003 dibangunkan dalam persekitaran Windows XP.

7.2.3.3.3 Kesimpulan Pengujian Prestasi

Berdasarkan kepada ketiga-tiga jenis pengujian prestasi yang dilakukan, didapati bahawa perisian Plagiarizer 2003 mempunyai prestasi yang baik kerana :

- Ia dapat memberikan satu tempoh masa tindakbalas yang munasabah kepada pengguna berdasarkan kepada spesifikasi komputer tersebut berdasarkan tahap bebanan yang dikenakan dalam melakukan proses analisa serta menjana laporan untuk mengesan dan menentukan peratusan plagiarisme
- Perisian Plagiarizer 2003 masih boleh beroperasi dalam keadaan optimum pada persekitaran yang melibatkan kes seperti pengujian kes 3 yang mana tahap bebanan adalah tinggi yang mana ia tidak menjadikan komputer tersebut tersangkut(*crash*)

- Perisian Plagiarizer 2003 masih boleh beroperasi dalam keadaan optimum pada persekitaran komputer yang rendah spesifikasi seperti Komputer C yang mana ia tidak menjadikan komputer tersebut tersangkut(*crash*)
- Perisian Plagiarizer 2003 serasi dengan platform Windows 98, Windows 98SE, Windows ME, Windows 2000 dan Windows XP

7.2.4 Pengujian Pemasangan

Pengujian pemasangan adalah bertujuan untuk memastikan Plagiarizer 2003 boleh berfungsi dan beroperasi dengan optimum pada persekitaran pengguna. Untuk tujuan tersebut, perisian Plagiarizer 2003 telah diuji pada persekitaran komputer pengguna yang berplatformkan Windows yang mana pengujian pemasangan merangkumi Windows XP, Windows 2000, Windows 98, Windows 98SE, dan Windows ME. Hasil daripada pengujian pemasangan di dapati Plagiarizer 2003 tidak mengalami masalah pemasangan cuma apa yang penting ialah proses pemasangan hendaklah dilakukan dengan teliti di mana komponen penting yang diperlukan seperti fail dll perlu dipasang ke dalam komponen Windows pengguna. Masalah penelitian ini diatasi dengan menggunakan alat pembangunan Install Shield for Microsoft Visual C++ yang membantu mengautomasikan proses pemasangan Plagiarizer 2003 ke atas komputer pengguna.

PENILAIAN PERISIAN

8.1 Pengenalan

Setelah Plagiarisme DITJ ditandatangani, suatu perjanjian akan dibuat ke atas perisian tersebut. Penilaian ini merupakan kemampuan perisian, kemampuan perisian, kelengkapan dan perancangan atau kadapan ke dalam perisian yang di...

8.2 Kekuatan Perisian Pihak

BAB 8

PENILAIAN PERISIAN

Plagiarisme DITJ ditandatangani, suatu perjanjian akan dibuat ke atas perisian tersebut. Penilaian ini merupakan kemampuan perisian, kemampuan perisian, kelengkapan dan perancangan atau kadapan ke dalam perisian yang di...

- Matrik untuk...
- Laporan...
- Identifikasi...

Oleh itu, kekuatan hukum Plagiarisme DITJ adalah seperti yang...

- Algoritma perisian...
- Kebolehan...

PENILAIAN PERISIAN

8.1 Pengenalan

Setelah Plagiarizer 2003 dibangunkan, suatu penilaian akan dibuat ke atas perisian tersebut. Penilaian ini merangkumi kekuatan perisian, kelemahan perisian, kekangan dan perancangan masa hadapan terhadap perisian yang dibangunkan.

8.2 Kekuatan Perisian Plagiarizer 2003

Plagiarizer 2003 telah dibangunkan dan direkabentuk mengikut kehendak pengguna melalui soal selidik yang telah dijalankan kepada para pensyarah iaitu berpaksikan kepada :

- Mudah untuk digunakan
- Laporan
- Masa tindakbalas yang munasabah

Oleh itu, kekuatan luaran Plagiarizer 2003 terletak kepada 3 perkara yang tersebut. Selain itu juga Plagiarizer 2003 mempunyai kekuatan dalaman dari segi :

- Algoritma pengesanan plagiarisme
- Kebolegunaan semula komponen

8.2.1 Kekuatan Luaran

8.2.1.1 Mudah Untuk Digunakan

Plagiarizer 2003 merupakan satu perisian automasi pengesanan plagiarisme yang mudah untuk digunakan. Matlamat pembangunannya adalah agar pengguna iaitu para pensyarah dan tutor tidak berasa takut atau janggal untuk menggunakannya selain Plagiarizer 2003 cuba mengelakkan daripada pengguna menjadi kecewa sekiranya mereka mengambil masa yang lama untuk mempelajari satu sistem dari memperoleh matlamat asal mereka. Maka pembangunan Plagiarizer 2003 cuba meminimakan fungsi-fungsi kompleks dan rumit agar pengguna tidak rasa bosan atau malas mempelajarinya.

Plagiarizer 2003 dikatakan mudah untuk digunakan kerana :

- 1) Ia membekalkan antaramuka berdasarkan piawai tettingkap Microsoft Windows yang menggunakan konsep *Graphical User Interface* (GUI) selain daripada menyokong sepenuhnya antaramuka WIMP (Windows, Icon, Menu, Pointer). Maka, dengan kata lain jika seseorang pengguna mempunyai pengalaman atau biasa menggunakan platform Windows maka satu jaminan boleh diberi bahawa pengguna tersebut yang ingin menggunakan Plagiarizer 2003 akan mengambil masa yang singkat untuk mempelajari Plagiarizer 2003.

- 2) Plagiarizer 2003 membenarkan pengguna membuat analisa secara asas untuk menentukan samada set tugas pelajar tersebut terlibat dengan plagiarisme atau tidak dengan hanya menggunakan konsep 3 butang sahaja iaitu *Import*, *Analyze* dan *Current Report* sahaja. Dengan konsep 3 butang ini, Plagiarizer 2003 mampu membuat analisa dan menjana laporan yang menunjukkan peratusan plagiarisme dan seterusnya terpulung kepada pensyarah atau tutor untuk memutuskan kes plagiarisme berdasarkan laporan yang dijana.

- 3) Plagiarizer membekalkan pengguna antaramuka yang mudah dengan wujudnya kekunci pantas (*Acceleration Key*) dan juga kekunci *Tab* bagi membenarkan pengguna hanya menggunakan papan kekunci sahaja di dalam kes di mana pengguna tidak mahu menggunakan tetikus dalam menggunakan perisian Plagiarizer 2003. Pertimbangkan di mana di dalam kes, jika berlakunya kerosakan ke atas papan kekunci pengguna atau tetikus pengguna bagaimana pengguna ingin menggunakan perisian Plagiarizer ini? Maka di sinilah letaknya kekuatan Plagiarizer 2003.

- 4) Plagiarizer 2003 membekalkan bar kemajuan(*Progress Bar*) yang menunjukkan kemajuan semasa Plagiarizer 2003 membuat analisa ke atas dokumen. Jika Plagiarizer 2003 tidak menunjukkan bar kemajuan maka pengguna tidak akan tahu samada perisian ni masih beroperasi atau tidak.

5) Plagiarizer 2003 membekalkan kotak amaran pengendali peristiwa dengan memaparkan deskripsi dan mesej ringkas yang sesuai apabila berlaku ralat yang dilakukan oleh pengguna misalnya jika pengguna mengklik butang *Analyze* tanpa mengimport sebarang fail maka, Plagiarizer 2003 akan mengeluarkan kotak amaran agar pengguna mengimport fail terlebih dahulu sebelum membuat analisa ke atas fail. Melalui cara ini, pengguna akan dapat mempelajari dan memahami bagaimana Plagiarizer 2003 berfungsi.

Kebolehgunaan Plagiarizer ini telah dibuktikan melalui kes pengujian kebolehgunaan yang telah dilakukan yang mana membuktikan bahawa secara puratanya pengguna baru yang menggunakan Plagiarizer 2003 hanya mengambil masa 8 minit sahaja untuk menguasai fungsi asas Plagiarizer 2003 tanpa bimbingan operator atau pengajar dan pengguna baru hanya mengambil masa 4 minit sahaja untuk menguasai fungsi asas Plagiarizer 2003 dengan bimbingan operator atau pengajar. Selain itu, melalui pengujian kebolehgunaan tersebut didapati bahawa kebanyakan pengguna atau responden berpuas hati dan memberi komen bahawa Plagiarizer 2003 mudah untuk digunakan kerana fungsinya yang mudah difahami dan tidak kompleks dan ini merupakan satu bukti petanda yang baik bahawa Plagiarizer 2003 mempunyai kebolehgunaan yang tinggi.

8.2.1.2 Laporan

Setiap kali proses analisa dibuat oleh pengguna untuk menentukan plagiarisme, Plagiarizer 2003 akan menjana 3 jenis laporan utama berasaskan fail HTML iaitu :

- 1) Laporan keseluruhan
- 2) Laporan berorientasikan graf
- 3) Laporan terperinci

Ketiga-tiga laporan ini sudah cukup untuk memberi panduan dan gambaran kepada pengguna iaitu para pensyarah atau tutor untuk mengesan kes plagiarisme yang berlaku dalam set tugas pelajar.

Laporan keseluruhan yang terdiri daripada seksyen Profil Analisa, Jadual Peratusan Plagiarisme dan Rumusan memberikan pengguna butiran keseluruhan mengenai set dokumen yang telah diuji dan peratusan plagiarisme dalam bentuk jadual. Selain itu, Plagiarizer 2003 turut menjana laporan yang berorientasikan graf yang menunjukkan secara grafik dokumen yang terlibat dengan plagiarisme. Selain itu juga Plagiarizer 2003 tidak hanya memberi laporan keseluruhan kasar sahaja, tetapi turut menunjukkan maklumat secara terperinci mengenai dokumen yang terlibat dengan Plagiarizer. Ini memberikan pengguna lebih banyak pilihan untuk melihat dan membaca laporan tersebut.

Tambahan, Plagiarizer 2003 membenarkan pengguna untuk mencetak laporan yang telah dijana selain turut membenarkan pengguna menyimpan laporan tersebut.

8.2.1.3 Masa Tindakbalas Analisa Yang Munasabah

Plagiarizer 2003 berkeupayaan membuat analisa ke atas set dokumen yang banyak dan memulangkan masa tindakbalas analisa yang munasabah dan singkat berdasarkan spesifikasi komputer pengguna. Ini telah dibuktikan berdasarkan kepada hasil keputusan pengujian prestasi yang telah dilakukan mengikut 3 tahap bebanan iaitu ringan, sederhana dan tinggi membuktikan bahawa Plagiarizer 2003 merupakan perisian automasi pengesanan plagiarisme yang cukup teguh dalam melakukan proses analisa. Jika masa tindakbalas analisa tersebut adalah munasabah dan singkat maka sudah tentu pengguna tidak merasa bosan untuk menunggu ataupun pengguna tidak berasa serik untuk menggunakan perisian Plagiarizer 2003 ini lagi akibat daripada kelembapan perisian. Tambahan dengan kemajuan perkakasan komputer masa kini, Plagiarizer 2003 tidak akan mengalami masalah kelembapan dalam membuat analisa.

8.2.2 Kekuatan Dalaman

8.2.2.1 Algoritma Pengesanan Plagiarisme

Selain itu juga kekuatan dalaman Plagiarizer2003 terletak pada pembangunan algoritma pengesanan plagiarisme. Plagiarizer menggunakan frekuensi bilangan perkataan (*word count frequency*) dalam mengesan samada suatu set dokumen yang diuji itu terlibat dengan plagiarisme atau tidak.

Algoritma frekuensi bilangan perkataan, boleh mengatasi masalah utama plagiarisme kerana teknik plagiarisme yang biasanya dilakukan oleh pelajar adalah dengan cara di mana :

- 1) Menyalin dan menyusun semula struktur ayat dan perenggan dengan cara mengubah kedudukan ayat atau perenggan tugas asal tersebut
- 2) Menyalin secara 100% daripada tugas asal tanpa membuat penukaran struktur ayat dan perenggan atau membuat penukaran ke atas perbendaharaan kata tugas kelas tersebut
- 3) Penggabungan dan pengurangan ayat atau perenggan daripada 2 atau lebih tugas asal

Di mana ini menunjukkan bahawa algoritma perisian yang dibangunkan perlu mengambilkira dan menganalisa keseluruhan bahagian fail tugas berbanding analisa secara bahagian demi bahagian bagi menentukan samada berlaku atau tidak plagiarisme dalam set tugas tersebut. Maka, di sinilah kelebihan yang terletak pada algoritma frekuensi bilangan perkataan kerana ia akan mengambilkira dan membuat analisa ke atas keseluruhan bahagian fail tugas.

Selain itu juga, untuk menambah ketepatan algoritma ini, Plagiarizer 2003 dilengkapi dengan fungsi pilihan penyelesaian iaitu :

- Mengabaikan tanda baca
- Mengabaikan nombor

Ini akan membenarkan pengguna membuat pilihan aturan penyelarasan bersesuaian dengan keadaan struktur fakta dokumen yang hendak diuji yang mana penyelarasan yang dibuat boleh meningkatkan lagi ketepatan analisa enjin algoritma.

8.2.2.2 Kebolegunaan semula

Oleh kerana Plagiarizer 2003, dibangunkan menggunakan bahasa berorientasikan objek iaitu C++, maka setiap komponen pengaturcaraan dipecahkan kepada kelas-kelas yang mana sudah tentu ia mempunyai konsep modulariti yang tinggi yang mana seterusnya meningkatkan tahap kebolegunaan semula kod sumber. Ini merupakan satu kekuatan dalaman Plagiarizer 2003 yang mana ini akan memudahkan pembangun perisian untuk menambah fungsi-fungsi baru kepada perisian pada masa akan datang dan memudahkan lagi kerja-kerja penyelenggaraan kod aturcara.

8.3 Kelemahan Perisian Plagiarizer 2003

Secara keseluruhannya, Plagiarizer 2003 telah dibangunkan dengan jayanya serta memenuhi objektif seperti yang telah dirancang pada fasa perancangan perisian dahulu. Namun begitu terdapat kelemahan-kelemahan pada Plagiarizer 2003 antaranya ialah:

- 1) Walaupun Plagiarizer 2003 merupakan perisian automasi untuk mengesan plagiarisme tetapi ia bukan perisian pintar kerana ia tidak dapat menentukan

siapa yang meniru siapa dalam kes plagiarisme yang berlaku. Hatta, pensyarah atau tutor sekalipun tidak dapat menentukan siapa yang meniru siapa jika terdapat kes di mana 2 dokumen pelajar yang lebih kurang sama. Walaubagaimanapun, ia adalah suatu yang tidak penting untuk menentukan siapa yang meniru siapa, kerana apa yang difokuskan ialah kedua-dua pelajar tersebut telah terlibat dengan plagiarisme.

2) Plagiarizer 2003 tidak dapat membuat analisa plagiarisme ke atas format dokumen selain daripada format *Word Documents (*.doc)*.

Tetapi ini merupakan suatu kelemahan yang kecil kerana kebiasaannya pada masa kini, tugas yang dibuat dan dihantar oleh pelajar adalah dalam format *Word Documents (*.doc)* yang berplatformkan *Office* dan ternyata kelemahan ini adalah minima.

3) Dalam laporan yang dijana oleh Plagiarizer 2003, ia tidak menunjukkan baris dan frasa mana yang terlibat dengan plagiarisme antara 2 dokumen yang telah dianalisa.

Plagiarizer 2003 tidak menyediakan fungsi sebegini kerana jika terdapat 100 tugas pelajar yang dianalisa dan setiap tugas tersebut terdiri daripada 30 mukasurat ia adalah tidak praktikal untuk menunjukkan semua baris dan frasa yang terlibat dengan plagiarisme pada antaramuka laporan dan ini merupakan sesuatu yang akan menyusahkan pengguna untuk melihat kesamaan antara frasa. Apa yang pengguna mahukan adalah suatu rumusan, graf, jadual dan maklumat yang menunjukkan potensi plagiarisme tersebut.

8.4 Kekangan Perisian Plagiarizer 2003

Secara keseluruhannya, Plagiarizer 2003 telah dibangunkan dengan jayanya serta memenuhi objektif seperti yang telah dirancang pada fasa perancangan perisian dahulu. Namun begitu terdapat kekangan-kekangan dalam menggunakan perisian Plagiarizer 2003 ini iaitu :

1) Prestasi

Prestasi analisa Plagiarizer 2003 bergantung kepada perkakasan komputer yang melarikan perisian ini. Masa tindakbalas analisa akan lebih pantas dan singkat jika unit pemprosesan pusat komputer tersebut tinggi dan rendah jika sebaliknya. Begitu juga dengan peruntukan memori. Plagiarizer 2003 memerlukan ruang ingatan yang cukup untuk membolehkan ia membuat peruntukan memori kepada set dokumen yang banyak. Namun begitu, jika Plagiarizer 2003 mendapati ruang ingatan tidak mencukupi sewaktu analisa, ia akan memaparkan mesej yang bersesuaian kepada pengguna. Kekangan prestasi merupakan lumrah bagi semua perisian dan apa yang penting saya telah meminimakan impak kekangan ini sewaktu pembangunan perisian Plagiarizer 2003

2) Hanya Untuk Dokumen Elektronik *Word Documents (*.doc)* sahaja

Untuk membuat analisa dan mengesan plagiarisme yang berlaku, Plagiarizer 2003 memerlukan tugas pelajar yang mana tugas tersebut perlu di dalam bentuk elektronik iaitu *softcopy* berbanding dengan *hardcopy*. Ini bermaksud setiap kali

penghantaran tugas, pelajar tersebut perlu menghantar *softcopy*, jika *hardcopy* sahaja dihantar analisa ke atas dokumen tersebut tidak dapat dilakukan. Selain itu juga, tugas pelajar tersebut perlu dalam format *Word Documents(*.doc)* kerana selain daripada format itu Plagiarizer 2003 mengekang pengguna daripada membuat analisa sebagai contohnya format *Adobe Acrobat(*.pdf)*.

3) Bergantung Kepada Pelayar Web(*Web Browser*)

Plagiarizer 2003 memerlukan pelayar web Internet Explorer untuk memaparkan laporan yang telah dijana. Jika pengguna tersebut tidak mempunyai pelayar web Internet Explorer maka laporan tidak dapat dipaparkan. Namun begitu, kekangan ini tidak menjadi suatu masalah kerana Plagiarizer 2003 akan membuat pemasangan pelayar web Internet Explorer kepada komputer pengguna untuk membenarkan pengguna melihat dan membaca laporan tersebut.

8.5 Cadangan Dan Perancangan Masa Hadapan

Walaupun perisian ini boleh dikatakan siap dan telah memenuhi objektif yang dikehendaki, diperkuatkan pula dengan tambahan-tambahan idea baru, namun, terdapat beberapa aspek yang boleh ditambah dan diperbaiki bagi menjadikan ini lebih teguh, mantap dan fleksibel.

Di antara aspek-aspek atau fungsi yang dirasakan perlu ditambah bagi perubahan di masa hadapan ialah antaranya :

- 1) Menjadikan Plagiarizer 2003 bukan sekadar perisian automasi sahaja tetapi ia merupakan satu perisian pintar (*intelligent software*) dengan menambah fungsi di mana Plagiarizer 2003 boleh membantu pensyarah atau tutor untuk mengesan atau mengenalpasti siapa yang meniru siapa jika wujudnya kes plagiarisme dalam set tugas pelajar
- 2) Plagiarizer 2003 boleh bekerja dan beroperasi dengan sistem pangkalan data berbanding penggunaan sistem fail untuk tujuan penyimpanan laporan agar perisian Plagiarizer 2003 menjadi lebih mantap dan teguh
- 3) Menaiktaraf fungsi Plagiarizer 2003 agar ia boleh mengesan plagiarisme yang berlaku di mana seorang pelajar membuat peniruan secara mentah atau 100% daripada Internet atau sumber lain selain daripada pelajar lain

- 4) Menambah fungsi-fungsi baru untuk penyalarsan agar analisa yang dibuat oleh enjin analisa perisian lebih tepat dalam menghasilkan laporan seperti contohnya mengambilkira kes huruf kecil dan huruf besar, mengambilkira atau mengabaikan perkataan hingar(*noise word*) seperti kata imbuhan, mengambilkira bilangan huruf dalam sesuatu perkataan dan lain-lain lagi fungsi yang boleh difikirkan untuk tujuan pembaikan perisian.
- 5) Kelajuan analisa Plagiarizer 2003 boleh ditingkatkan lagi dengan cara memperbaiki enjin analisa dalam kod aturcara agar prestasi Plagiarizer 2003 menjadi lebih pantas dalam membuat analisa ke atas set tugasn pelajar yang banyak.
- 6) Laporan yang dijana oleh Plagiarizer 2003 boleh diperbaiki lagi dengan memberi atau menunjukkan baris, ayat dan frasa yang terlibat dengan plagiarisme dalam set dokumen tugasn pelajar yang diuji.
- 7) Plagiarizer 2003 boleh diperluaskan lagi fungsinya dalam menentukan kes plagiarisme dengan boleh mengesan plagiarisme yang berlaku dalam set tugasn pengaturcaraan iaitu fail-fail yang berformat *.c, *.cpp, *.prolog.

8.6 Kesimpulan

Perisian Plagiarizer 2003 pada dasarnya telah berjaya memenuhi matlamat dan objektif pembangunannya iaitu untuk membantu para pensyarah dan tutor untuk mengesan kadar peniruan atau kegiatan plagiarisme di kalangan pelajar. Saya yakin berdasarkan algoritma yang dibangunkan ini, Plagiarizer 2003 mampu mengesan kes plagiarisme yang berlaku di kalangan pelajar. Cuma apa yang penting ialah pemantauan secara berterusan perlu dilakukan agar kelemahan-kelemahan yang wujud dan timbul boleh diperbaiki dari semasa ke semasa. Jika satu usaha bersama serta penyelidikan dijalankan secara bersepadu sudah tentu perisian ini dapat dikomersilkan pada masa akan datang kerana masih belum banyak lagi perisian seumpama ini digunakan di institusi-institusi pengajian tinggi di Malaysia. Diharapkan agar Plagiarizer 2003 dapat memberi manfaat kepada warga tenaga pengajar bagi mengurangkan kes plagiarisme yang berlaku di kalangan pelajar.

SENARAI RUJUKAN

Davis, M. (1993). *Software Requirements: Objects, Functions and States*. Prentice Hall.

Dezuan Bahasa dan Pustaka (1994). *Alfama Devon, Etila Keliga, Dewap Bahasa dan Pustaka*

Finlay, S. (1995). *CaseCrest*. Dharma rajana: University of Birmingham.

SENARAI RUJUKAN

Hasty, B. (1997). Steve Jobs: The director's new film is a \$10 million program suit, but so's history. *Free Press All Time Magazine*

Diperolehi dari World Wide Web

<http://www.time.com/time/magazine/1997/05/mag97124/0,9167,40767,00.html>

Hog, H., McKown, B. (1985). The decomposition of human-computer secondary activities. In *Proceedings of SIGCHI '85*, 129-136

Lepelate, Seymour. (2007). *Software Outline of Theory and Problems of Essential Computer Mathematics*. McGraw-Hill.

Paul, C. (2000). *Analysis of natural and programming languages: an overview of various tools and languages*. Department of Computer Science, University of Sheffield

Diperolehi dari World Wide Web

http://www.dcs.shef.ac.uk/~c100/teaching/HT00/HT00_Version_1

Rice, J., Shari Lawrence. (1996). *Software Engineering Theory & Practice International Edition*. Prentice Hall.

Seymour, J. (1986). *Software Engineering Fifth Edition*. Addison Wesley.

SENARAI RUJUKAN

Davis, M.(1993). *Software Requirements: Objects, Functions and States*. Prentice Hall.

Dewan Bahasa dan Pustaka.(1994). *Kamus Dewan*. Edisi Ketiga. Dewan Bahasa dan Pustaka.

Finlay, S.(1999). *CopyCatch*. Disertasi sarjana. University of Birmingham.

Handy, B.(1997). Steven Stealberg? The director's new film is hit with a \$10 million plagiarism suit, but isn't history free to all. *Time Magazine*.150(22).

Diperolehi dari World Wide Web :

http://www.time.com/time/magazine/1997/dom/97124/the_arts_show.steven_stealb.html

Jing, H., McKeown, R. (1999). The decomposition of human-written summary sentences. *In Proceedings of SIGIR'99*.129,136

Lipshutz, Seymour. (2000). *Schaum's Outline of Theory and Problem of Essential Computer Mathematics*. Mc Graw-Hill.

Paul, C.(2000). *Plagiarism in natural and programming languages: an overview of current tools and technologies*. Department of Computer Science, University of Sheffield

Diperolehi dari World Wide Web :

[\(http://www.dcs.shef.ac.uk/~cloughie/plagiarism/HTML_Version/ \)](http://www.dcs.shef.ac.uk/~cloughie/plagiarism/HTML_Version/)

Pfleeger, Shari Lawrence.(1998). *Software Engineering Theory & Practice International Edition*. Prentice Hall.

Sommerville, Ian.(1996). *Software Engineering Fifth Edition*. Addison Wesley.

Woolls, D.(1999). PAPERS: authors, affiliations and abstract. *IAFL Conference '99*.
Diperolehi daripada World Wide Web :
(http://www-clg.bham.ac.uk/forensic/IAFL99/conf_abst.html).

LAMPIRAN

APENDIKS

Universiti Malaya

**LAMPIRAN
&
APENDIKS**

Universiti Malaya

QUESTIONNAIRE

Borang Soal Selidik

Dr. Nur Hafizah Binti Md. Nordin
Mohd Rizal B. Kadir
Faculty of Computer Science & Information Technology
University of Malaya
Kuala Lumpur

Under The Supervision Of
Associate Professor Razli Anwar Zuberah Bt. Raja Zainal



QUESTIONNAIRE

We are in gratitude and would like to say thank you for your co-operation for this questionnaire. Currently, we are making a research to **develop software to detect plagiarism in students' assignment** to fulfill our **Final Project Phase I (WXES3181)**. Therefore the purpose of this questionnaire is to gain the information about the plagiarism case in students' assignment where it is a very important to assist in capturing the requirement to develop this software. Hopefully, it is highly recommended and appreciated that your party could provide a good co-operation in favor of completing this particular questionnaire. Thus, any comment or suggestions based on the questionnaire are highly acceptable.

Prepared by:

Mohd Hairul Nizam B. Md. Nasir

Mohd Rizal B. Kadis

Faculty of Computer Science & Information Technology

University of Malaya

Kuala Lumpur

Under The Supervision Of:

Associates Professor Raja Ainon Zabariah Bt. Raja Zainal

General Section

**Please indicate your answers by ticking \surd in the provided boxes*

1) I think that the proportion of students involved in a typical outbreak of plagiarism in students' assignment is:

- More than 76%
- Between 51% and 75%
- Between 26% and 50%
- Between 10% and 25%
- Less than 10%

2) I think that the problem of plagiarism in students' assignment:

- Not a problem
- A minor nuisance
- Under control
- Bad and getting worse

3) Do you or does your department routinely check all submitted assignment for indications of copying?

- Yes (please answer question 4 and ignore 5)
- No (please answer question 5 and ignore 4)

4) The checking system used is:

- Non-automated, by hand and eye (manual)
 - A public service or software
 - A service or software developed and/or operated in-house
 - A part of an integrated teaching environment
 - Others (please specify)
-



5) A checking system is not used because:

- Nobody has ever started to use one
 - A decision has been taken not to use one
 - It is believed that no or little cheating takes place
 - The group sizes are so small it is not needed
 - Others (please specify)
-

6) I think that the **major** reason why students resort to plagiarism is:

- Assignments that have been given are too difficult for them
 - They are too disorganized to complete the work in time
 - There is too much pressure from other subjects or family
 - Others (please specify)
-

7) What is the **major** type of plagiarism that have been used by students to complete their assignment?

- Copy and paste from their peer/other students' assignment
 - Copy and paste from the internet such as paper mills, essay bank
 - Others (please specify)
-

8) I think that, software to detect plagiarism in students' assignment is:

- Crucial needed
 - Not needed (please specify why)
-



Students' Assignment: Natural Languages Types

- 9) What is commonly students' assignment file type if the softcopy needed to be submitted other than word documents format (*.doc)?

**Please number 1 to 4 according to the priority for the question 10, where number 1 indicates the highest priority and number 4 indicates the lowest priority*

- 10) What is the **technique of plagiarism** that have been used by the students to complete their assignment?

Description	Priority order
1. Sentence or paragraph structure reordering	_____
2. Sentence or paragraph rewording (paraphrasing)	_____
3. Sentence or paragraph reduction	_____
4. Sentence or paragraph combination	_____

If there are any others technique please specify,

- 11) Can you explain **thoroughly** about this plagiarism technique for the most students used to complete their assignment based on your answer above?



Students' Assignment: Programming Languages Types

12) Have you taught any subject of programming languages?

- Yes (Please answer questions 13 & 14)
 No (Skip questions 13 & 14)

**Please number 1 to 3 according to the priority for the question 13, where number 1 indicates the highest priority and number 3 indicates the lowest priority*

13) In your opinion what type of programming languages does most of the students resort to plagiarism to complete their laboratory assignment during laboratory session?

Languages

Priority Order

1. C

2. C++

3. Java

If there are any others please specify,

**Please number 1 to 5 according to the priority for the question 14, where number 1 indicates the highest priority and number 5 indicates the lowest priority*

14) What is the technique of plagiarism that have been used by the students to complete the laboratory assignment?

Description

Priority Order

1. Change and modify variable name

2. Change and modify control structure

3. Changes and modify data type

4. Change or delete comments

5. Change and modify identifier

If there are any others please specify,



Software Functionality Expectation

15) If I use software to detect plagiarism in students' assignment, the factor I am to consider is :

Description	Priority order
1. Ease to use (user friendly)	_____
2. High performance with minimal response time	_____
3. Accuracy of report	_____
4. Cost effectiveness	_____

If there are any others factor please specify,

16) What the content or type of the report do you expect to display a statistical information if plagiarisms exist in students' assignment?

- Percentage between the assignment and details paragraph or phrase involved
- Comparison table and details paragraph or phrase involved
- Others (please specify)

17) Do you need the print function to print the report that indicates the plagiarism in students' assignment?

- Yes
- No





Borang Soal Selidik

Sila tandakan \surd pada kotak yang disediakan bagi jawapan yang bersesuaian

- 1) Pernahkah anda terlibat dengan gejala plagiarisme dalam menyiapkan tugas yang diberikan?
 - Pernah (Jawab ke semua soalan di bawah)
 - Tidak Pernah (Tinggalkan ke semua soalan di bawah)

- 2) Nisbah plagiarisme yang dilakukan oleh anda dalam 10 tugas diberikan ialah :
 - 1 hingga 2 tugas
 - 3 hingga 4 tugas
 - 5 hingga 6 tugas
 - 7 hingga 8 tugas
 - 9 hingga 10 tugas

- 3) Pada pendapat anda, faktor utama anda terlibat dengan gejala plagiarisme ini ialah kerana :
 - Tugas yang telah diberikan adalah sukar
 - Tidak dapat menguruskan masa dan kerja dengan sempurna
 - Terlalu banyak tekanan daripada subjek/kursus lain
 - Lain-lain (Sila nyatakan) _____

- 4) Jenis utama plagiarisme yang dilakukan oleh anda dalam menyiapkan tugas yang telah diberikan ?
 - Mengambil dan menyalin semula daripada tugas pelajar yang lain
 - Mengambil dan menyalin daripada internet
 - Lain-lain (Sila nyatakan)

- 5) Apakah jenis fail tugas anda jika salinan disket tugas perlu dihantar selain daripada format word document(*.doc) ?



Soalan Temubual Pelajar



- 1) Pernahkah anda terlibat dengan gejala plagiarisme dalam menyiapkan tugas yang diberikan?
- 2) Apakah jenis plagiarisme yang dilakukan oleh anda dalam menyiapkan tugas yang telah diberikan?
- 3) Apakah teknik plagiarisme yang anda gunakan supaya dapat mengelakkan daripada ditangkap kerana terlibat dengan gejala plagiarisme ini?
- 4) Sila huraikan dengan lebih lanjut bagi teknik yang anda nyatakan seperti di atas?
- 5) Apakah jenis fail tugas anda jika salinan disket tugas perlu dihantar selain daripada format word document (*.doc) ?

Sila nyatakan sebarang komen atau cadangan jika ada.

1.3 Mengira Peratusan Plagiarisme Melalui Kuantiti Perbandingan Perbandingan Kesamaan Bilangan Perkataan Antara Dua Teks

Keseluruhan ini secara umum bermaksud untuk mengira peratusan plagiarisme antara

2 naskah dengan cara mengira peratusan bilangan perkataan yang sama antara

2 naskah tersebut. Keseluruhan ini akan lebih baik jika dilakukan dengan

perbandingan yang sama antara 2 naskah yang sama antara 2

naskah tersebut. Keseluruhan ini akan lebih baik jika dilakukan dengan

perbandingan antara 2 naskah yang sama antara 2 naskah yang sama antara 2

naskah yang sama antara 2 naskah yang sama antara 2 naskah yang sama antara 2

naskah yang sama antara 2 naskah yang sama antara 2

Algoritma Pengiraan Peratusan Plagiarisme Plagiarizer 2003

Melalui keseluruhan ini, peratusan antara 2 naskah yang sama antara 2 naskah yang sama antara 2

keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

Menggunakan keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

keseluruhan ini akan lebih baik jika dilakukan dengan perbandingan yang sama antara 2

keseluruhan ini

Algoritma Pengiraan Peratusan Plagiarisme Plagiarizer 2003

1.0 Mengira Peratusan Plagiarisme Melalui Kaedah Peratusan Frekuensi Kesamaan Bilangan Perkataan Antara Dua Tugas

Kaedah ini secara asasnya bertujuan untuk mengira peratusan plagiarisme antara 2 tugas dengan cara mengira peratusan frekuensi bilangan perkataan yang sama antara 2 tugas tersebut. Konsepnya mudah di mana, lebih banyak frekuensi bilangan perkataan yang sama antara 2 tugas, maka lebih tinggi peratusan plagiarisme antara 2 tugas tersebut. Di sini, mekanisme yang digunakan ialah berdasarkan konsep pengiraan analisa 2 tugas demi 2 tugas yang bermaksud sistem akan membuat analisa ke atas 2 fail tugas dahulu dan setelah selesai sistem akan membuat analisa ke atas 2 fail tugas yang lain pula.

Melalui kaedah ini, plagiarisme antara 2 tugas dapat dikenalpasti dalam kes-kes di mana plagiarisme dilakukan dengan cara mengubah struktur posisi atau kedudukan seperti yang dinyatakan di bawah yang mana kes-kes ini merupakan teknik-teknik plagiarisme utama atau yang biasa dilakukan oleh para pelajar dalam menyiapkan tugas mereka dengan masa yang singkat iaitu :

- Mengubah struktur ayat atau perenggan dalam tugas mereka di mana ayat atau perenggan asal diubah kedudukannya berbanding dengan tugas asal

- Melakukan penggabungan dan pengurangan ayat

Kaedah pengiraan peratusan plagiarisme tersebut adalah seperti dinyatakan seperti di bawah:

Diberi 2 tugas X dan Y masing-masing, peratusan plagiarisme dapat dikira berdasarkan frekuensi bilangan perkataan yang sama dengan :

- 1) Mengira frekuensi bilangan perkataan yang sama iaitu n_{XY} antara 2 tugas iaitu tugas X dan tugas Y dan juga jumlah bilangan perkataan bagi setiap tugas X dan Y.
- 2) Untuk mengira peratusan plagiarisme bagi tugas X dan Y masing-masing formula pengiraan adalah seperti di bawah iaitu :

$$\text{Peratusan plagiarisme bagi tugas, \%} = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Dalam Tugas}}$$

Contohnya, untuk mengira:

Peratusan plagiarisme tugas X terhadap tugas Y ialah :

$$\text{Peratusan plagiarisme bagi tugas, \%}_X = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Tugas X}}$$

dan

Peratusan plagiarisme tugas X terhadap tugas Y ialah :

$$\text{Peratusan plagiarisme bagi tugas, } \%_Y = \frac{\text{Bilangan Perkataan Yang sama, } n_{XY}}{\sum \text{Bilangan Perkataan Tugas Y}}$$

Contoh sampel kes plagiarisme yang berlaku :

Tugas X mempunyai jumlah bilangan perkataan sebanyak 75, 000 patah perkataan dan tugas Y mempunyai jumlah bilangan perkataan sebanyak 50, 000 patah perkataan dan bilangan perkataan yang sama antara dua tugas adalah 42, 000 patah perkataan, maka berdasarkan kes ini :

Peratusan plagiarisme bagi tugas X kepada tugas Y ialah :

$$\frac{42,000}{75,000} * 100 = 56.0\%$$

Peratusan plagiarisme bagi tugas Y kepada tugas X ialah :

$$\frac{42,000}{50,000} * 100 = 84.0\%$$

2.0 Implementasi Algoritma Dalam Dunia Sebenar

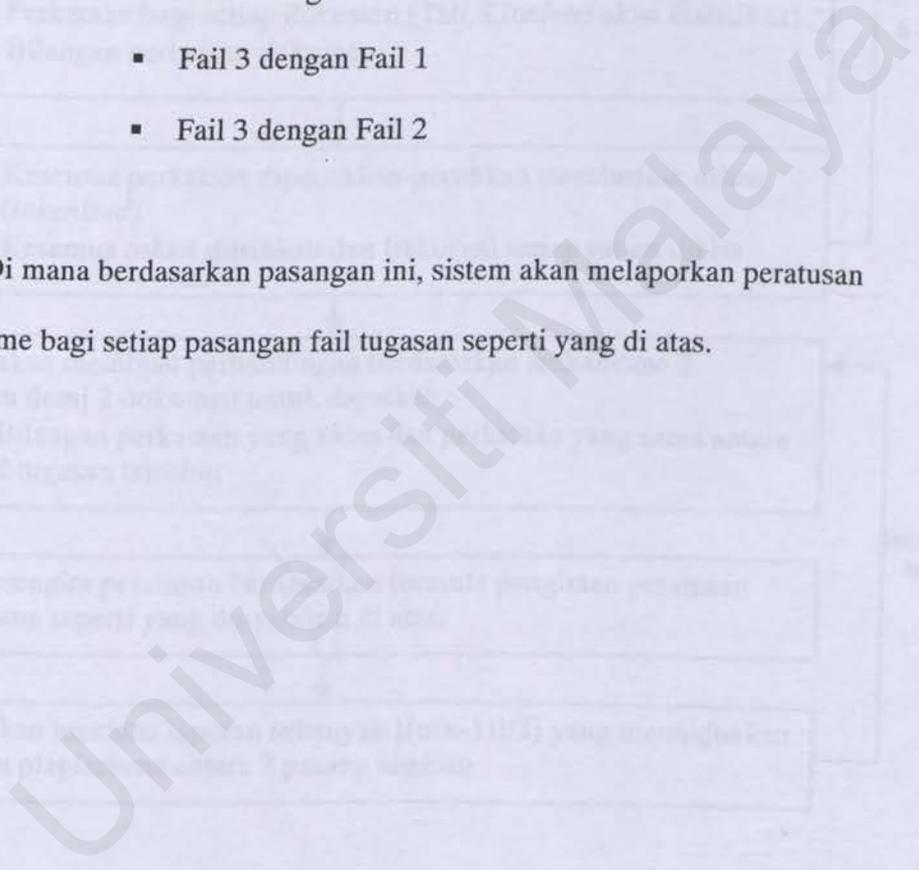
Dalam dunia sebenar, pengguna akan membuat analisa ke atas lebih daripada 2 fail tugas. Di sini akan ditunjukkan bagaimana sistem membuat analisa ke atas 3 fail tugas sebagai sampel contoh iaitu Fail 1, Fail 2 dan Fail 3 berdasarkan kepada mekanisme 2 tugas demi 2 tugas. Langkah-langkah adalah seperti berikut :

- 1) Sistem akan membaca dan mengekstrak maklumat daripada ketiga-tiga fail tugas tersebut bagi mendapatkan maklumat mengenai jumlah bilangan perkataan bagi setiap fail tugas tersebut.
- 2) Kesamaan perkataan antara 2 fail tugas akan didapatkan iaitu merujuk kepada sampel ini sistem akan mencari bilangan kesamaan perkataan bagi 3 pasangan fail iaitu bilangan pasangan tersebut mestilah $[(n(n-1))/2]$ pasangan di mana n adalah bilangan yang hendak diuji iaitu :
 - Pasangan Fail 1 dengan Fail 2
 - Pasangan Fail 1 dengan Fail 3
 - Pasangan Fail 2 dengan Fail 3
- 3) Kemudian sistem akan membuat pengiraan peratusan plagiarisme berdasarkan mekanisme 2 tugas demi 2 tugas untuk menunjukkan peratusan tersebut iaitu dengan menggunakan formula $[n(n-1)]$ untuk menentukan bilangan pasangan fail tugas yang perlu di analisa di mana n merujuk kepada bilangan

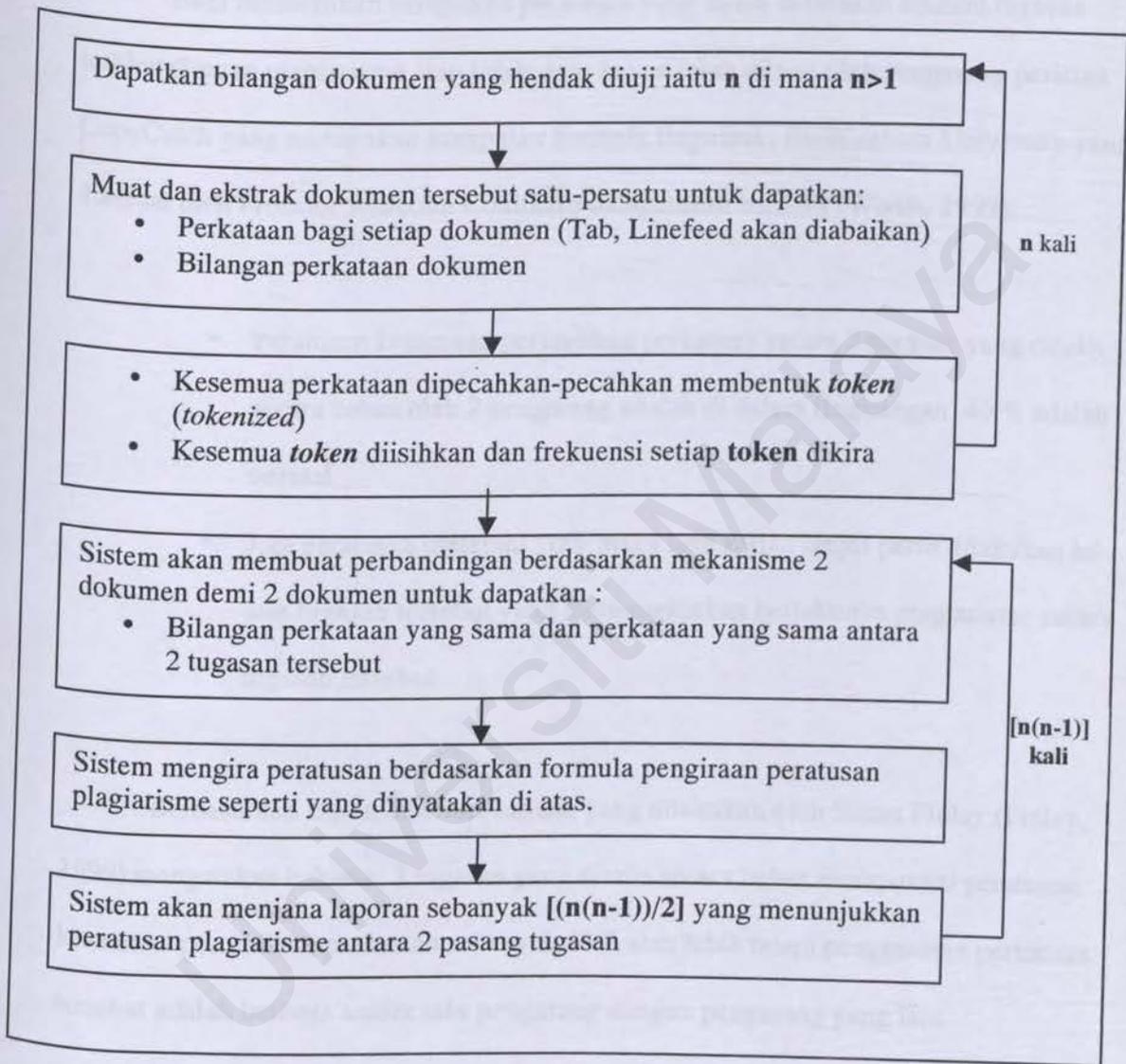
tugasan yang hendak diuji, maka dalam konteks ini bilangan pasangan tugas yang mesti ialah 6 pasang iaitu :

- Fail 1 dengan Fail 2
- Fail 1 dengan Fail 3
- Fail 2 dengan Fail 1
- Fail 2 dengan Fail 3
- Fail 3 dengan Fail 1
- Fail 3 dengan Fail 2

Di mana berdasarkan pasangan ini, sistem akan melaporkan peratusan plagiarisme bagi setiap pasangan fail tugas seperti yang di atas.



Rajah 2 : Gambaran Diagrama Plagiarisme 2013 Melalui Analisis Pengiraan



Rajah 2 : Gambaran Bagaimana Plagiarizer 2003 Membuat Analisa Pengesanan Plagiarisme

3.0 Menentukan Tahap Peratusan Plagiarisme

Bagi menentukan berapakah peratusan yang boleh dikatakan sesuatu tugas terlibat dengan plagiarisme atau tidak, satu kajian telah dibuat oleh pengarang perisian CopyCatch yang merupakan kumpulan forensik linguistik, Birmingham University yang diketuai oleh Profesor Malcolm Coulthard mengatakan bahawa (Wools, 1999):

- Peratusan kesamaan pertindihan perkataan antara 2 tugas yang ditulis secara bebas oleh 2 pengarang adalah di dalam lingkungan 40 % adalah normal
- Jika peratusan melebihi 70% maka satu kajian lanjut perlu dilakukan ke atas tugas tersebut yang memungkinkan berlakunya plagiarisme antara tugas tersebut.

Berdasarkan kajian disertasi sarjana yang dilakukan oleh Susan Finlay (Finlay, 1999) mengatakan bahawa, 2 tugas yang ditulis secara bebas mempunyai peratusan kesamaan pertindihan perkataan sebanyak 50% atau lebih tetapi penggunaan perkataan tersebut adalah berbeza antara satu pengarang dengan pengarang yang lain.

Oleh itu berdasarkan kepada 2 kajian yang dijalankan di atas saya menetapkan bahawa peratusan plagiarisme antara set tugas adalah seperti dalam jadual 1 di bawah iaitu :

Tahap Lingkungan Peratusan	Deskripsi
0% hingga 40%	Merupakan satu peratusan yang bebas daripada plagiarisme
41% hingga 70%	Merupakan satu peratusan yang normal
71% hingga 100%	Merupakan satu peratusan yang memungkinkan berlakunya plagiarisme antara tugas dan kajian terperinci perlu dilakukan ke atas tugas tersebut.

Jadual 1: Penentuan Tahap Peratusan Plagiarisme

Tetapi peratusan plagiarisme ini adalah terpulang kepada pengguna iaitu para pensyarah untuk menentukan samada tugas tersebut terlibat atau tidak dengan plagiarisme berdasarkan peratusan yang dijana oleh sistem perisian penganalisa plagiarisme ini.

Manual Pengguna

Manual Pengguna

Perisian Pencegahan Plagiarisme
Plagiarism 2003

Universiti Malaya

Copyright © 2003 MAMPU (MAMPU) Sdn. Bhd.
Copyright © Fakulti Sains Komputer & Teknologi Maklumat, Universiti Malaya

Untuk maklumat lanjut, sila rujuk ke www.mampu.com

Manual Pengguna



Plagiarizer 2003

*Perisian Penganalisa Plagiarisme
Plagiarizer 2003*

Copyright © 2003 Mohd Hairul Nizam B. Md. Nasir

Copyright © Fakulti Sains Komputer & Teknologi Maklumat, Universiti Malaya

Untuk maklumat lanjut sila e-mel ke semangat_daulat@hotmail.com

Senarai Isi Kandungan

Senarai Isi Kandungan	2
Perisian Plagiarizer 2003	4
1.0 Pemasangan Perisian Plagiarizer 2003	5
1.1 Keperluan Sistem	5
1.2 Proses Pemasangan	6
2.0 Menggunakan Perisian Plagiarizer 2003	8
2.1 Sejenak Menggunakan Plagiarizer	8
2.2 Mula	9
2.3 Mengimport Fail Dokumen	10
2.3.1 Tip dan Syor : Mengimport Fail Dokumen	11
2.4 Membuat Aturan Analisa & Laporan	12
2.4.1 Penerangan	13
2.4.1.1 Aturan Analisa(<i>Analyze Rules</i>)	13
2.4.1.2 Aturan Berkaitan Laporan (<i>Report Related Settings</i>)	14
2.4.2 Tip dan Syor : Aturan Analisa & Laporan	15
2.5 Menyingkirkan Dokumen dalam Senarai Perpustakaan Dokumen Plagiarizer	17
2.6 Menganalisa Fail Dokumen	18
2.7 Melihat Laporan	19
2.7.1 Laporan Semasa	19
2.7.2 Laporan Lepas	20

2.7.3	Jenis Laporan yang Dijana	20
2.7.3.1	Laporan Keseluruhan	20
2.7.3.2	Laporan Berorientasi Graf	22
2.7.3.3	Laporan Terperinci	23
2.8	Mengosongkan Medan Senarai Laporan	24
3.0	Penyingkiran Perisian Plagiarizer 2003	25
4.0	Penyelesaian Masalah dan Bantuan Pengguna	26

Plagiarizer 2003 adalah program antarmuka pengguna yang tidak memerlukan pengetahuan mendalam mengenai teknologi. Berfungsi sebagai alat untuk menganalisis dan menilai laporan yang dihasilkan oleh alat pemeriksa keseluruhan untuk mengesan plagiarisme yang berlaku dalam dokumen pengguna. Plagiarizer 2003 berkeupayaan menganalisis laporan yang dihasilkan oleh alat pemeriksa keseluruhan dan menunjukkan tempat dan plagiat yang berlaku dalam artikel, bab, subbab dan dokumen kepeluasan laporan.

Kelebihan Plagiarizer 2003 yang diperkenalkan oleh alat pemeriksa keseluruhan adalah kemudahan dalam penggunaan yang komprehensif dalam melakukan analisis penggunaan plagiat. Selain itu, ia berkemampuan untuk menganalisis dokumen yang panjang yang tidak dapat dibaca yang singkat.

Oleh itu, Plagiarizer 2003 merupakan satu pilihan perisian yang baik yang tepat dan komprehensif yang boleh digunakan oleh ahli sebagai panduan, pemalar atau alat untuk mengesan gejala plagiarisme yang berlaku dalam dokumen para pelajar berdasarkan laporan yang dijana oleh Plagiarizer 2003.



Plagiarizer 2003 merupakan perisian automasi yang dibangunkan untuk para pensyarah dan tutor bagi menganalisa dan mengesan plagiarisme dalam set tugas pelajar. Ia telah direkabentuk berpaksikan kepada 3 konsep utama iaitu:

- **Mudah untuk digunakan**
- **Berketepatan tinggi**
- **Berprestasi tinggi**

Plagiarizer 2003 membekalkan antaramuka yang mudah agar anda tidak merasa takut atau janggal untuk menggunakannya. Berbekalkan kepada 3 konsep mudah iaitu **import**, **analisa** dan **lihat laporan** sudah cukup mampu untuk membenarkan anda untuk mengesan plagiarisme yang berlaku dalam tugas pelajar. Plagiarizer 2003 berkeupayaan menjana laporan yang berorientasikan jadual, graf dan rumusan yang menunjukkan peratusan plagiarisme bagi memudahkan anda membaca, melihat dan mencetak keputusan laporan tersebut.

Rekabentuk Plagiarizer 2003 yang mengambilkira ciri-ciri ketepatan menjadikannya satu perisian automasi yang komprehensif dalam membuat analisa pengesanan plagiarisme. Selain itu, ia berkemampuan membuat analisa dalam set tugas yang banyak dalam masa yang singkat.

Oleh itu, Plagiarizer 2003 merupakan satu pilihan perisian automasi yang tepat dan komprehensif yang boleh digunakan oleh anda sebagai pensyarah, pengajar atau tutor untuk mengesan gejala plagiarisme yang berlaku dalam tugas para pelajar berdasarkan laporan yang dijana oleh Plagiarizer 2003.

1.0 Pemasangan Perisian Plagiarizer 2003

1.1 Keperluan Sistem

Plagiarizer 2003 telah direkabentuk untuk platform komputer berasaskan Windows.

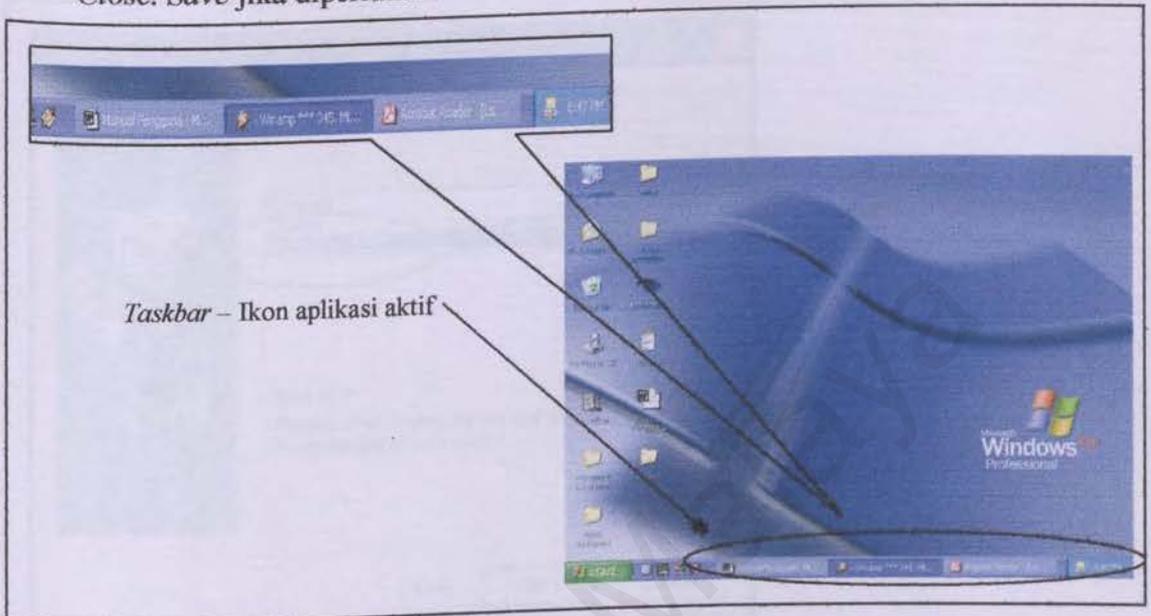
Keperluan sistem berasaskan Windows :

- ✓ Komputer peribadi untuk Windows 98, 98SE, ME, 2000*, XP
- ✓ Unit Pemprosesan pusat 233Mhz atau lebih baik
- ✓ 32MB sedia ada untuk ingatan capaian rawak RAM (128MB untuk Windows XP)
- ✓ 5MB sedia ada untuk ruang cakera keras atau lebih baik
- ✓ Pemacu cakera padat
- ✓ Skrin monitor dengan resolusi minimum 800 x 600 piksel.
- ✓ Pencetak untuk fungsi cetakan laporan

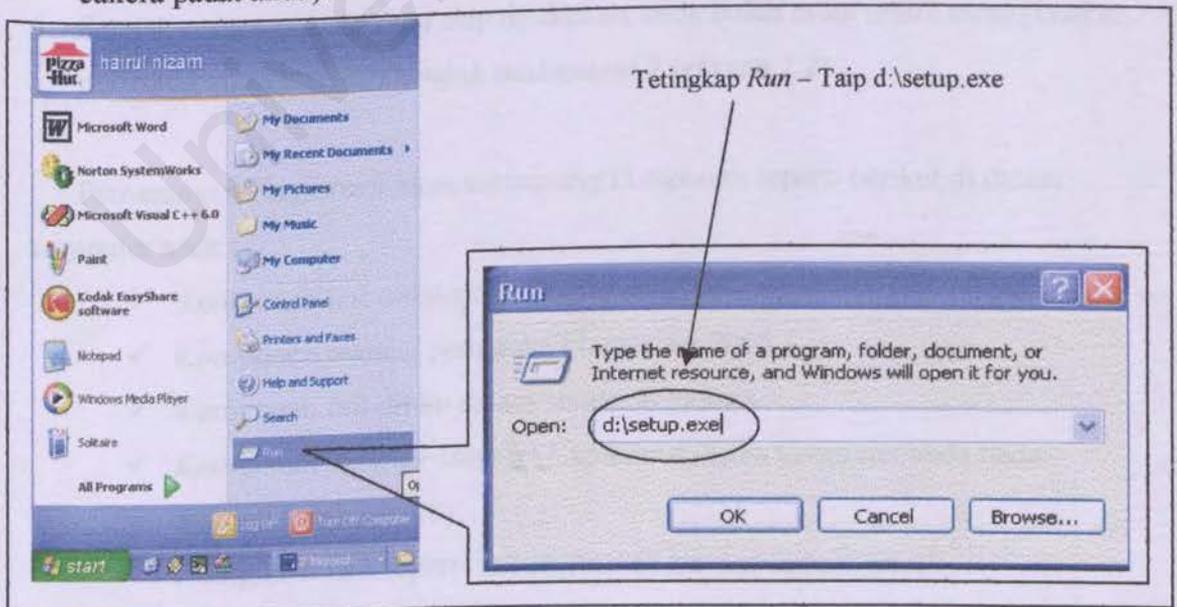
** Untuk Windows 2000, anda mesti mempunyai hak istimewa sebagai pentadbir (administrator) untuk memasang perisian ini.*

1.2 Proses Pemasangan

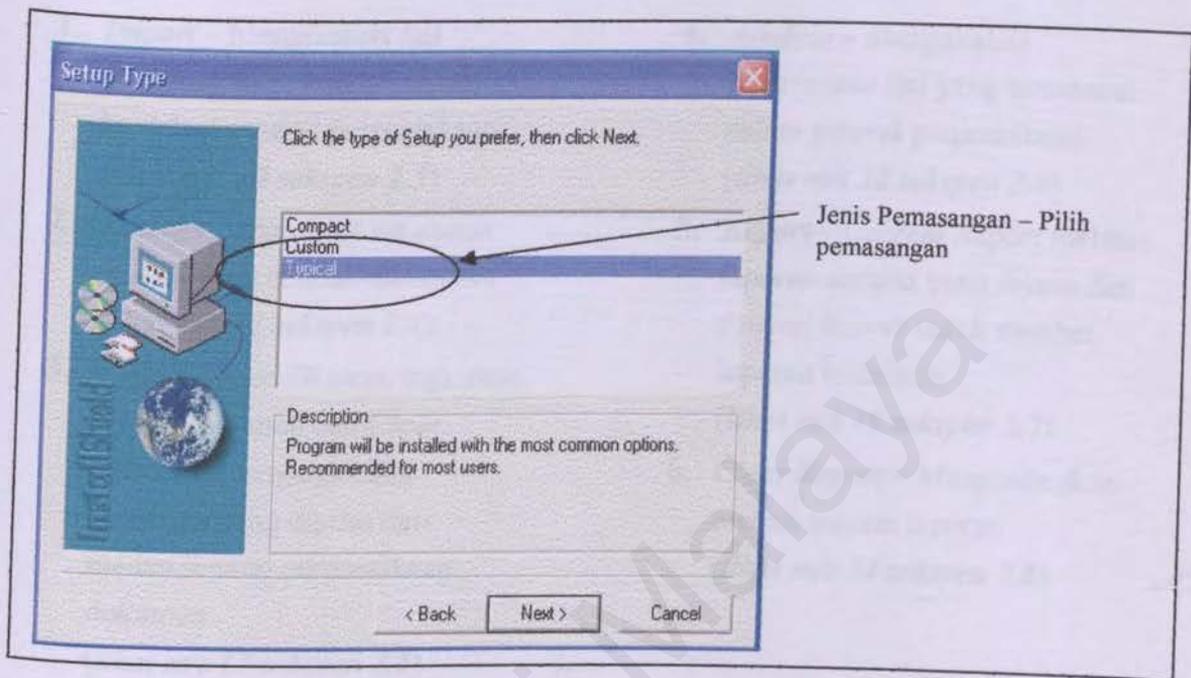
1. Tutup semua aplikasi aktif. Pada *taskbar*, klik-kanan setiap ikon aplikasi. Pilih *Close*. *Save* jika diperlukan.



2. Masukkan cakera padat perisian Plagiarizer 2003 ke dalam pemacu cakera padat.
3. Jika tetingkap pemasangan tidak muncul dalam beberapa saat, pilih *Run* daripada menu *Start* dan taip `d:\setup.exe` (jika `d` adalah merupakan *letter* untuk pemacu cakera padat anda).



4. Ikut segala arahan pada skrin. Dicadangkan agar anda memilih pemasangan *typical* untuk pemasangan Plagiarizer 2003.



5. Walaubagaimanapun anda boleh membatalkan proses pemasangan ini dengan mengklik pada butang *Cancel* dan butang *Back* untuk kembali kepada tettingkap sebelumnya.
6. Setelah proses pemasangan siap dilakukan, anda boleh mula untuk menggunakan perisian Plagiarizer 2003 (*rujuk mukasurat 9 seksyen 2.2*).

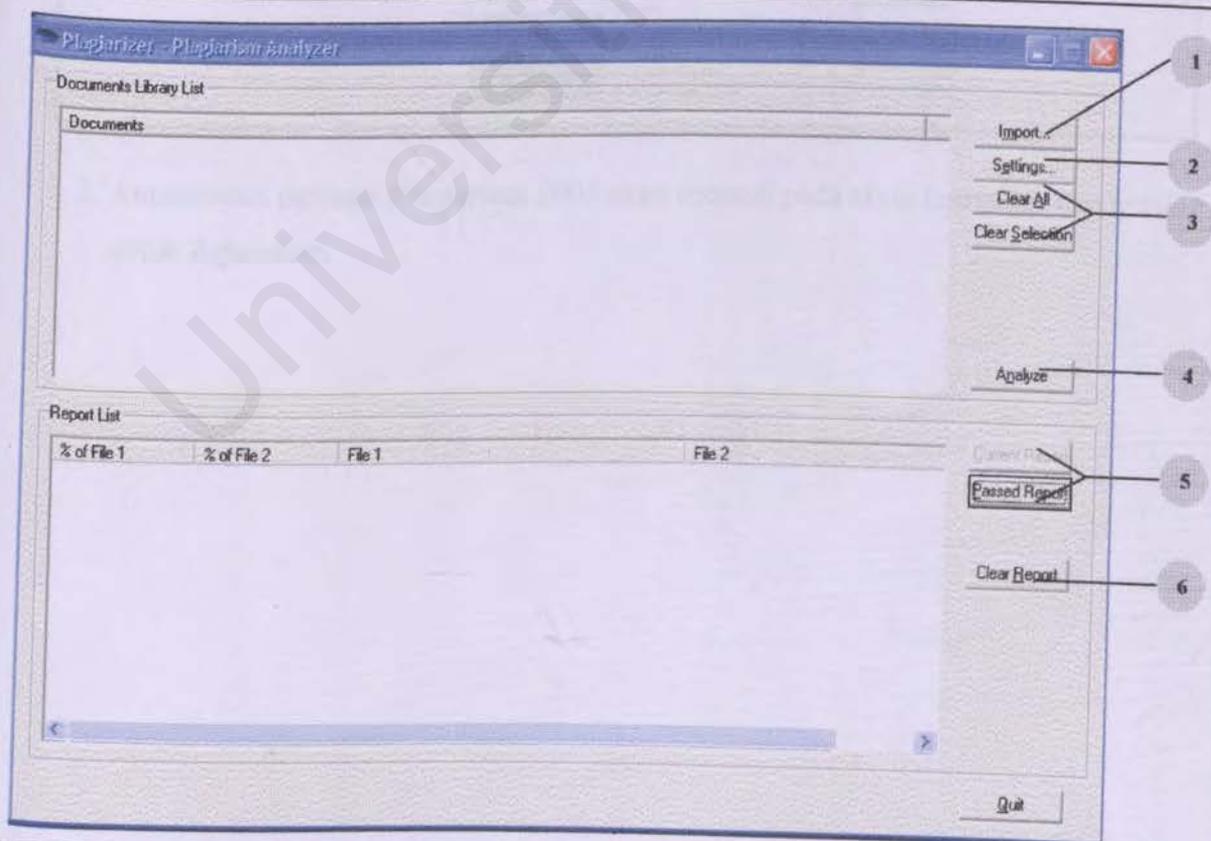
Pemasangan Plagiarizer akan memasang komponen seperti berikut di dalam komputer anda :

- ✓ Komponen fail terlaksana Plagiarizer 2003
- ✓ Komponen manual pengguna Plagiarizer 2003
- ✓ Komponen fail dll ke dalam sistem Windows
- ✓ Komponen *browser* Internet Explorer 4.0 (jika komputer anda tiada Internet Explorer 4.0)
- ✓ Komponen lain seperti fail README.txt dan fail *uninstall*

2.0 Menggunakan Perisian Plagiarizer 2003

2.1 Sejenak Menggunakan Plagiarizer 2003

1. **Import** – Mengimport fail dokumen yang hendak dianalisa ke dalam senarai perpustakaan (lihat m/s 10 seksyen 2.3)
2. **Settings** – Membuat set aturan analisa yang hendak dilakukan (lihat m/s 12 seksyen 2.4)
3. **Clear** – *Clear All* menyingkirkan semua dokumen dan *Clear Selection* menyingkirkan dokumen yang dipilih dari medan senarai perpustakaan dokumen (lihat m/s 17 seksyen 2.5)
4. **Analyze** – Menganalisa plagiarisme fail yang tersenarai dalam senarai perpustakaan (lihat m/s 18 seksyen 2.6)
5. **Report** – *Current Report* melihat laporan semasa yang dijana dan *Passed Report* untuk melihat laporan terdahulu (lihat m/s 19 seksyen 2.7)
6. **Clear Report** – Mengosongkan medan senarai laporan (lihat m/s 24 seksyen 2.8)



2.2 Mula

1. Klik pada menu *Start* pada *taskbar*. Pergi ke menu *All Programs*. Pilih menu *Plagiarizer*. Klik pada ikon *Plagiarizer 2003*.

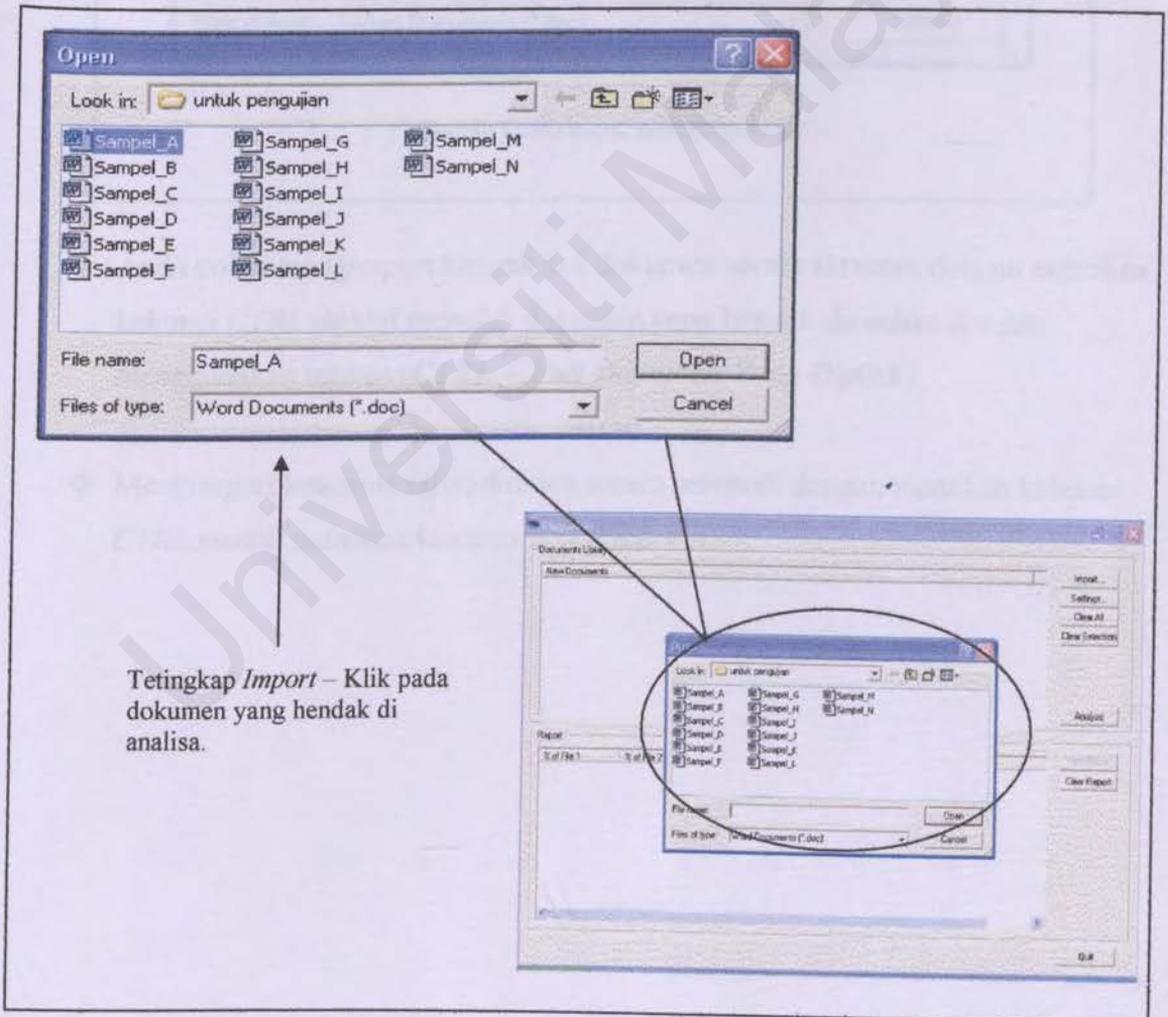


2. Antaramuka perisian Plagiarizer 2003 akan muncul pada skrin komputer dan sedia untuk digunakan.

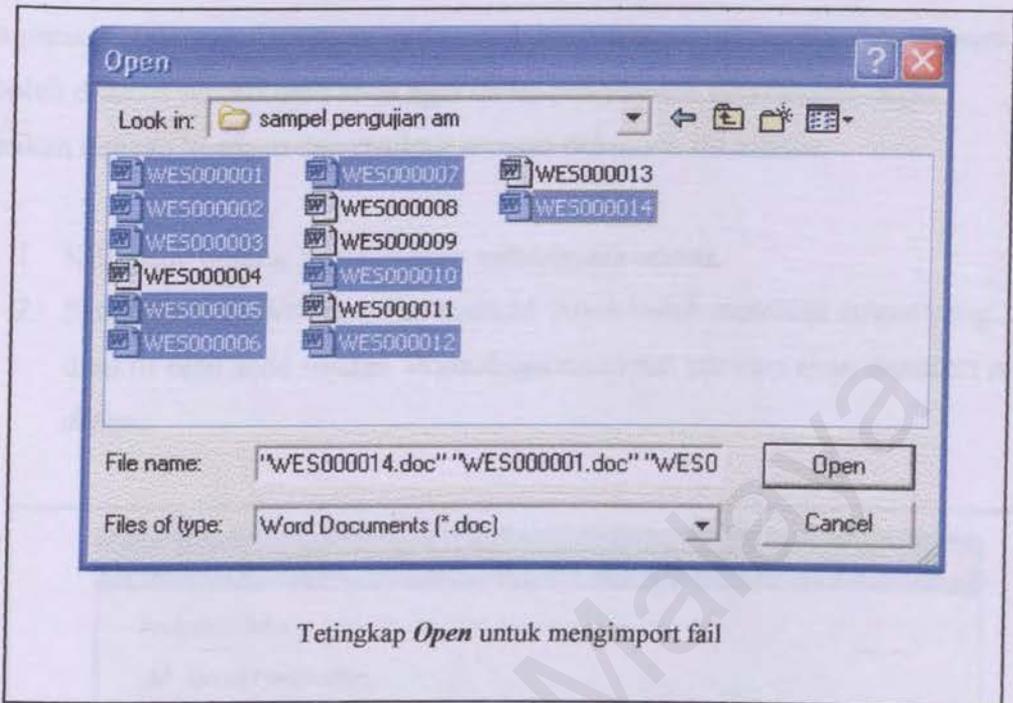
2.3 Mengimport Fail Dokumen

Untuk menganalisa dokumen yang dikehendaki, anda perlu mengimport fail dokumen terlebih dahulu ke dalam senarai perpustakaan dokumen Plagiarizer 2003.

1. Klik pada butang *Import* pada antaramuka utama..
2. Satu tetingkap *Import* akan muncul. Cari lokasi di mana letaknya dokumen yang hendak diuji.
3. Pilih dokumen yang hendak di analisa.
4. Klik dua kali pada ikon dokumen tersebut atau klik pada butang *open* untuk dimasukkan ke dalam senarai perpustakaan dokumen Plagiarizer 2003.



2.3.1 Tip dan Syor : Mengimport Fail Dokumen

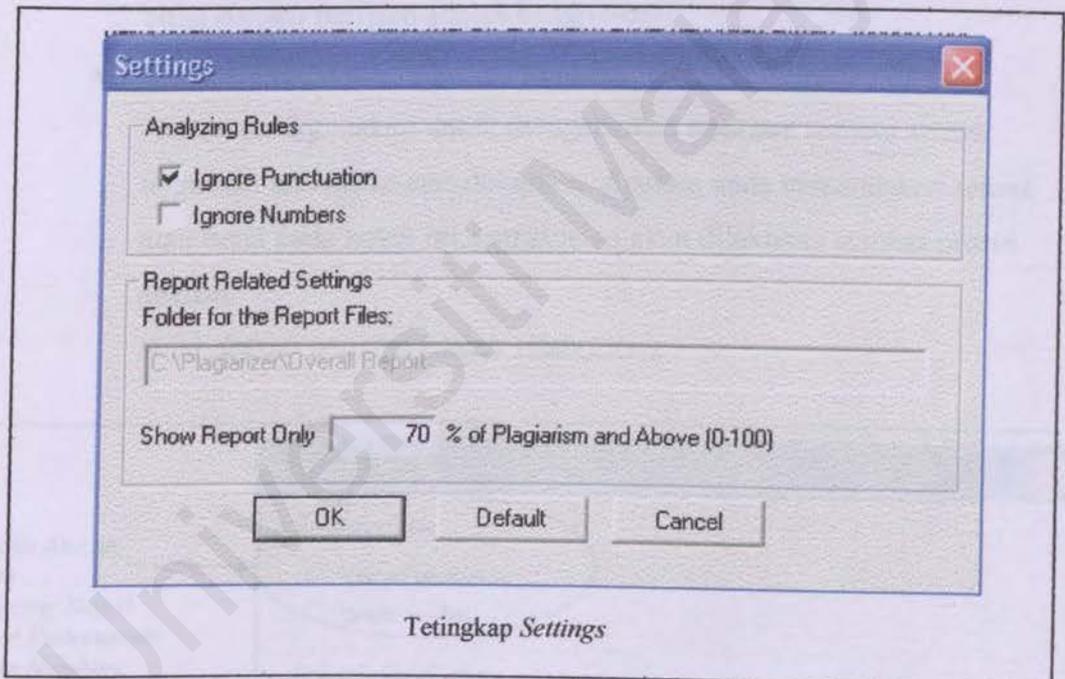


- ❖ Anda boleh mengimport banyak fail dokumen secara serentak dengan menekan kekunci *CTRL* sambil memilih dokumen yang hendak dianalisa dengan menggunakan tetikus (*CTRL + Fail Dokumen Yang Dipilih*)
atau;
- ❖ Mengimport kesemua fail dokumen secara serentak dengan menekan kekunci *CTRL* sambil menekan kekunci A (*CTRL + A*)

2.4 Membuat Aturan Analisa dan Laporan

Plagiarizer 2003 membenarkan anda untuk membuat aturan analisa dan laporan yang boleh ditakrif sendiri oleh anda agar tahap pengesanan plagiarisme dapat disesuaikan dengan keadaan dan struktur semasa dokumen itu sendiri.

1. Klik pada butang *Settings* pada antaramuka utama.
2. Satu tettingkap *Settings* akan muncul. Anda boleh membuat aturan yang ditakrif oleh anda sendiri. Walaubagaimanapun perisian akan memberi nilai *default*.



2.4.1 Penerangan

2.4.1.1 Aturan Analisa (*Analyzing Rules*)

Terdapat 2 kotak semak iaitu :

- ***Ignore Punctuation***

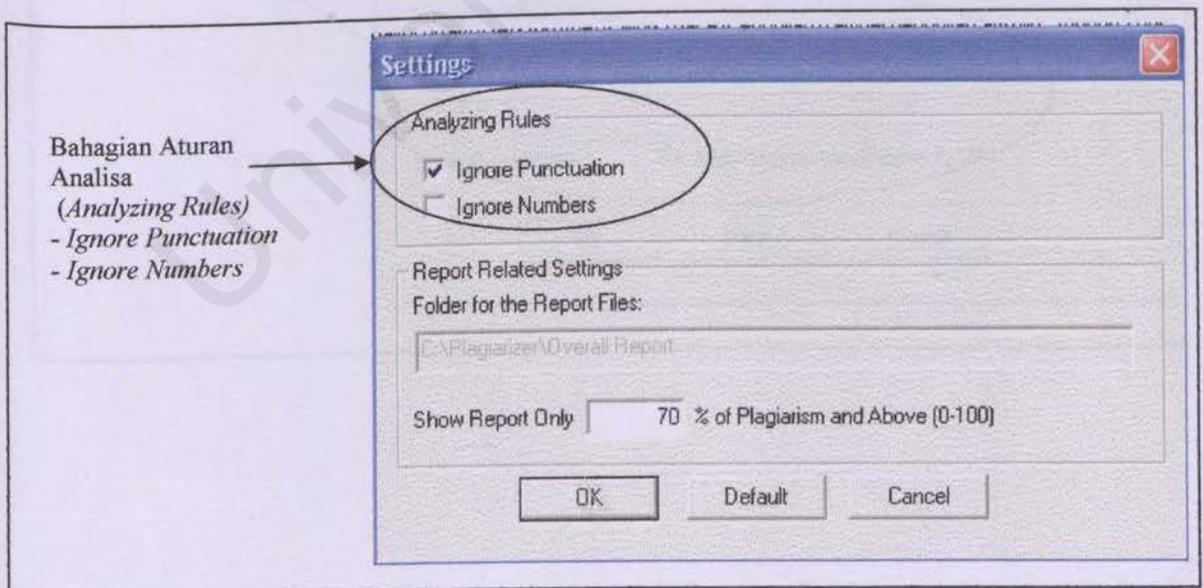
Aturan ini digunakan untuk mengabaikan **tanda baca** semasa sistem membuat analisa ke atas dokumen. Apabila anda menandakan semak atau betul pada kotak ini, pengabaian akan dilakukan semasa proses analisa.

Nilai *default* perisian adalah benar(*true*).

- ***Ignore Numbers***

Aturan ini digunakan untuk mengabaikan **nombor** semasa sistem membuat analisa ke atas dokumen. Apabila anda menandakan semak atau betul pada kotak ini, pengabaian akan dilakukan semasa proses analisa.

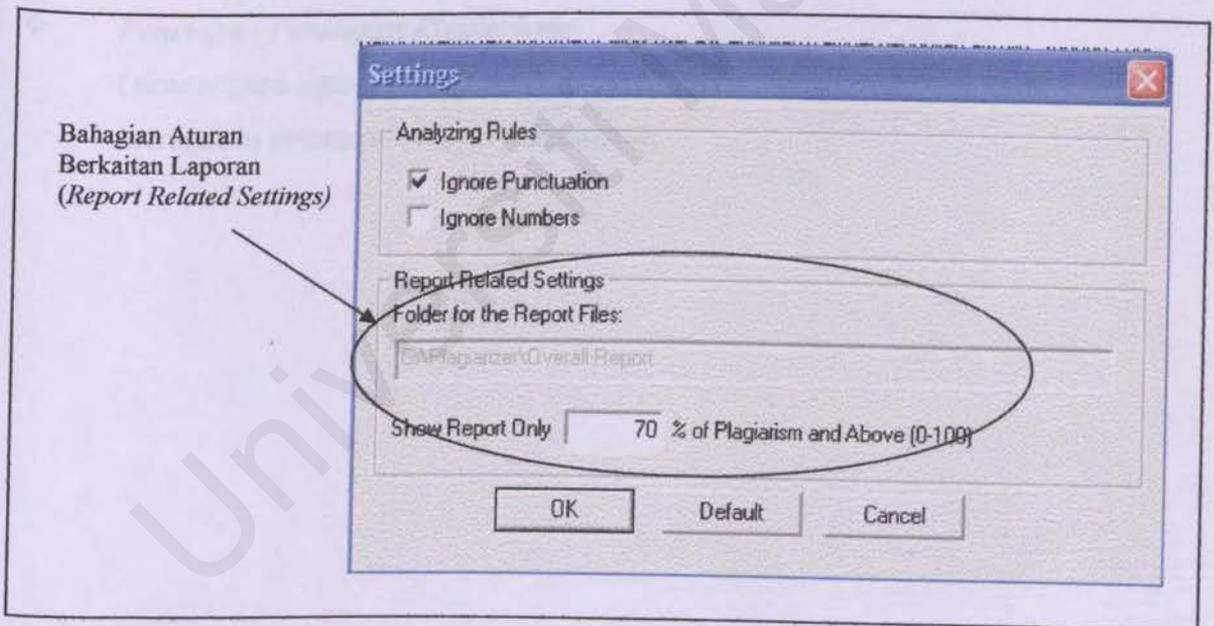
Nilai *default* perisian adalah salah(*false*).



2.4.1.2 Aturan Berkaitan Laporan (*Report Related Settings*)

Terdapat 2 bahagian iaitu:

- **Bahagian Pertama :** Medan ruangan teks tidak aktif menunjukkan kepada anda di mana laporan yang dijana di mana nilai *default* perisian adalah di dalam direktori `c:\Plagiarizer\Overall Report`
- **Bahagian Kedua :** Anda boleh menetapkan peratusan plagiarisme di mana sistem hanya menjana laporan yang akan menunjukkan dokumen yang terlibat dengan plagiarisme yang sama atau melebihi peratusan ini sahaja. Nilai *default* sistem adalah 70% dan peratusan minimum dan maksimum yang dibenarkan adalah 0%,99% masing-masing.



2.4.2 Tip dan Syor : Aturan Analisa dan Laporan

Dicadangkan di sini agar anda mengikut nilai awal(*default value*) yang telah ditetapkan oleh perisian Plagiarizer 2003

❖ ***Ignore Punctuation***

Dicadangkan agar anda menyemak kotak semak untuk bahagian ini agar potensi pengesanan plagiarisme adalah lebih tepat.

❖ ***Ignore Numbers***

Dicadangkan agar anda tidak menyemak kotak semak untuk bahagian ini agar potensi pengesanan plagiarisme adalah lebih tepat **kecuali** dalam kes di mana analisa dibuat ke atas dokumen yang mana nombor menjadi asas fakta

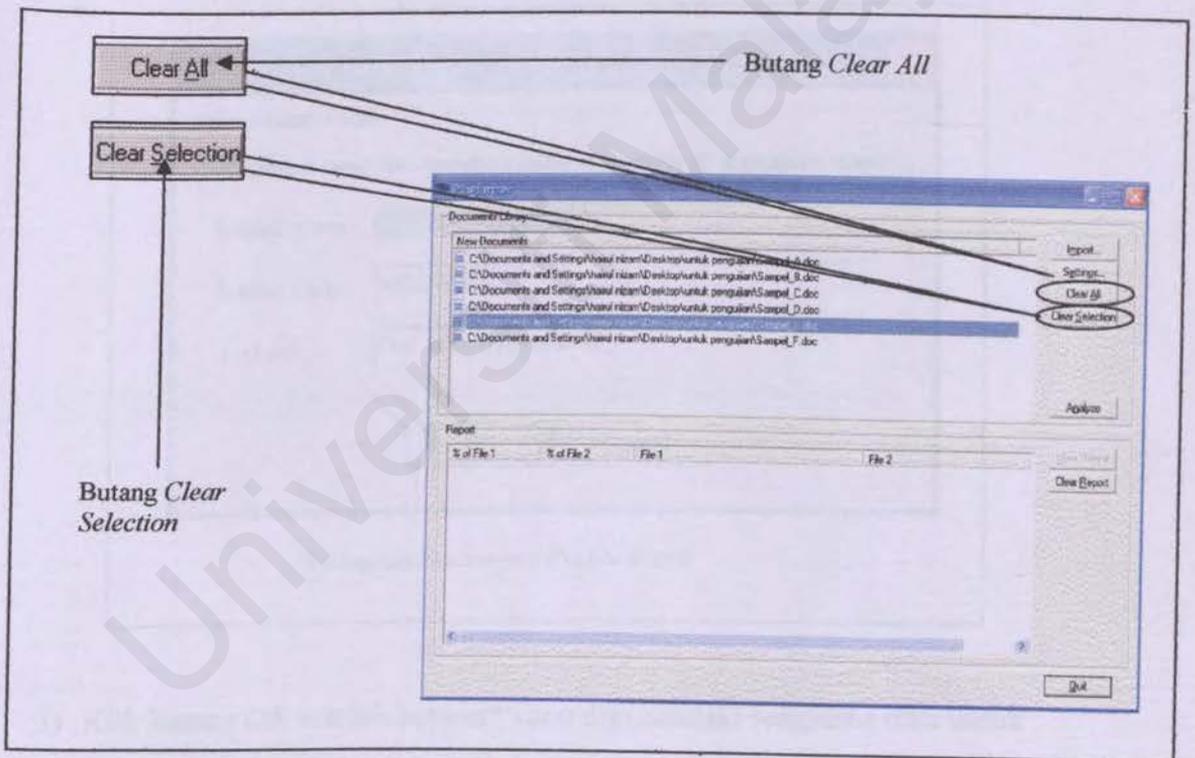
❖ ***Penetapan Peratusan Plagiarisme***

Dicadangkan agar menetapkan peratusan plagiarisme kepada 70% kerana ini merupakan peratusan potensi plagiarisme.

2.5 Menyingkirkan Dokumen dalam Senarai Perpustakaan Dokumen Plagiarizer

Plagiarizer 2003 membenarkan anda untuk menyingkirkan senarai dokumen yang terdapat di dalam senarai perpustakaan dokumen Plagiarizer. Ini dapat dilakukan dengan 2 cara :

1. Klik pada butang *Clear All* pada antaramuka utama untuk menyingkirkan semua senarai dokumen yang terdapat di dalam medan senarai perpustakaan Plagiarizer.
2. Klik pada butang *Clear Selection* pada antaramuka utama untuk menyingkirkan senarai dokumen yang dipilih oleh anda sahaja.

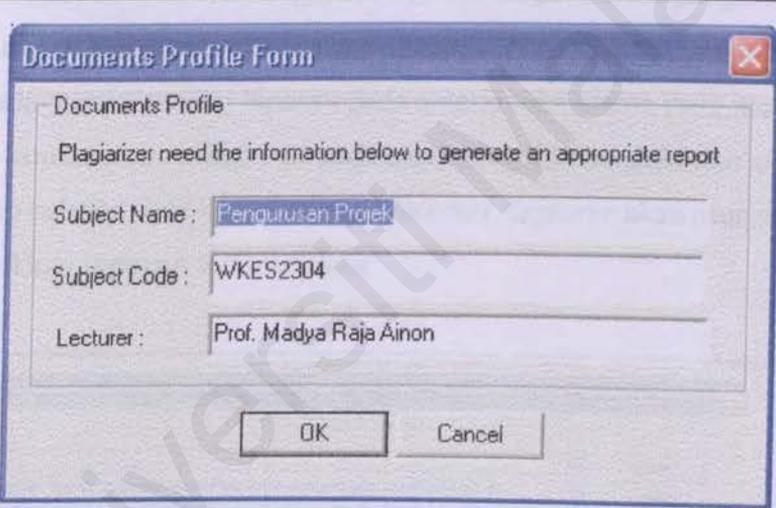


2.6 Menganalisa Fail Dokumen

Anda hanya perlu klik pada butang *Analyze* untuk melakukan analisa ke atas dokumen yang terdapat di dalam senarai perpustakaan tetapi sebelum itu Plagiarizer 2003 akan meminta anda mengisi butiran penting yang diperlukan agar laporan yang dijana adalah sempurna.

Untuk menganalisa :

- 1) Klik pada butang *Analyze* pada antaramuka utama
- 2) Isi butiran yang berkaitan dalam tettingkap *Documents Profile Form* iaitu nama subjek, kod subjek dan nama pensyarah.



The image shows a screenshot of a software dialog box titled "Documents Profile Form". The dialog box has a title bar with a close button (X) in the top right corner. Inside the dialog, there is a section titled "Documents Profile" with the instruction "Plagiarizer need the information below to generate an appropriate report". Below this instruction are three text input fields: "Subject Name" containing "Pengurusan Projek", "Subject Code" containing "WKES2304", and "Lecturer" containing "Prof. Madya Raja Ainon". At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

Tetingkap *Documents Profile Form*

- 3) Klik butang *OK* setelah butiran* yang dikehendaki sempurna diisi untuk membenarkan proses analisa dilakukan atau butang *Cancel* untuk membatalkan proses analisa.

* Butiran hendaklah diisi dengan tepat agar laporan yang dijana oleh Plagiarizer 2003 adalah sempurna dan tepat.

2.7 Melihat Laporan

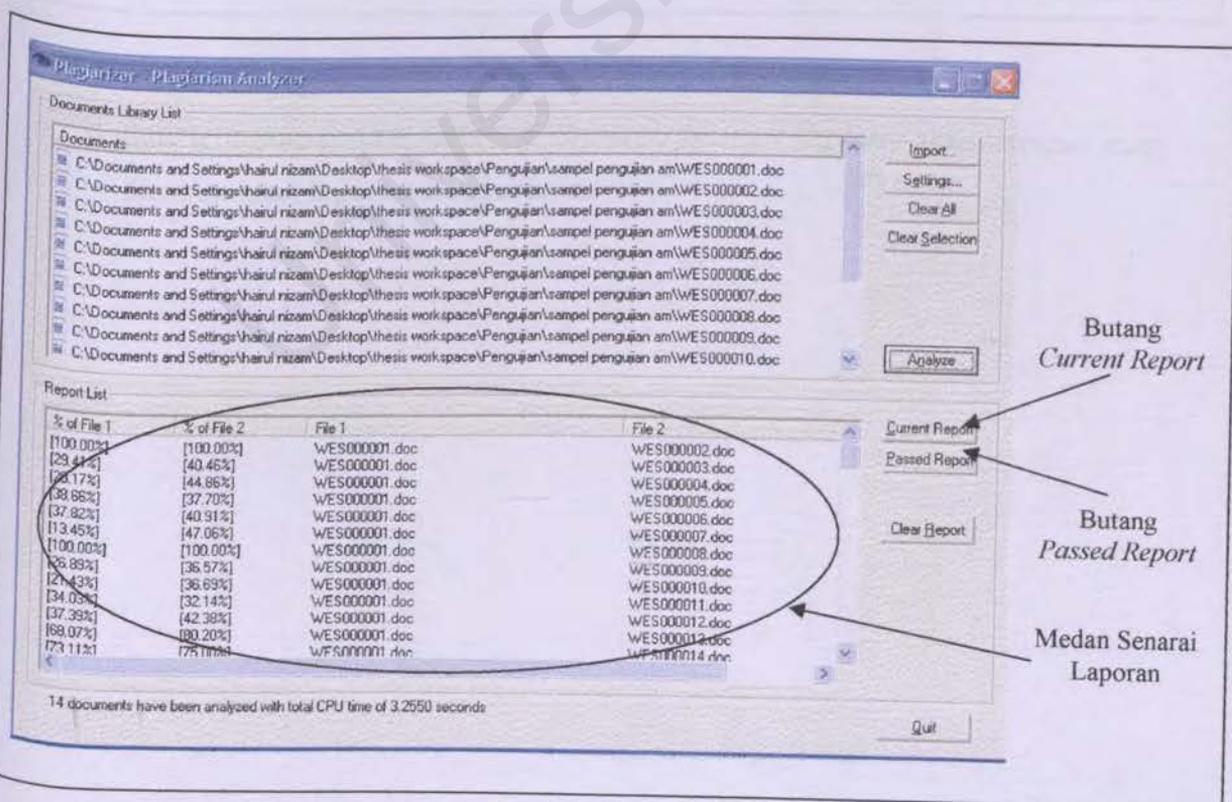
Anda boleh melihat laporan yang telah dijana iaitu:

- 1) Laporan semasa iaitu laporan yang dijana pada kali terakhir anda mengklik butang *Analyze*
- 2) Laporan lepas iaitu laporan yang telah dijana pada masa atau waktu dahulu

2.7.1 Laporan Semasa

Laporan semasa boleh dilihat dengan dua cara iaitu:

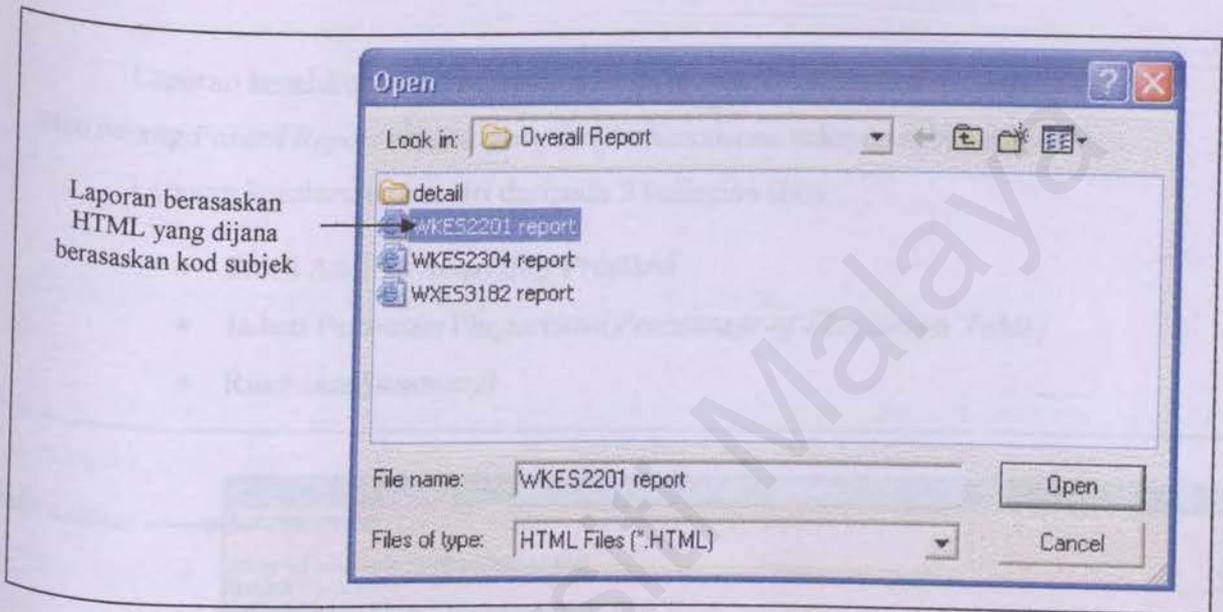
- 1) Klik pada butang *Current Report* pada antaramuka utama dan satu tettingkap pelayar web (*Web Browser*) *Internet Explorer* akan muncul yang menunjukkan **laporan keseluruhan atau;**
- 2) Lihat pada medan senarai laporan pada antaramuka utama yang menunjukkan laporan keseluruhan, klik 2 kali pada baris yang dikehendaki dan satu tettingkap pelayar web (*Web Browser*) *Internet Explorer* akan muncul yang menunjukkan **laporan keseluruhan**



2.7.2 Laporan Lepas

Laporan lepas boleh dilihat dengan :

- 1) Klik pada butang *Passed Report* pada antaramuka utama
- 2) Satu tetingkap *open* akan terpapar
- 3) Pilih dengan klik pada laporan berasaskan fail HTML yang hendak dilihat berdasarkan kod subjek di dalam *Folder Overall Report*



- 4) Klik butang *open*
- 5) Satu tetingkap pelayar web (*Web Browser*) *Internet Explorer* akan terpapar yang menunjukkan **laporan keseluruhan**

2.7.3 Jenis Laporan yang Dijana

Plagiarizer 2003 menjana 3 jenis laporan :

- 1) Laporan Keseluruhan
- 2) Laporan Berorientasi Graf
- 3) Laporan Terperinci

2.7.3.1 Laporan Keseluruhan

Laporan keseluruhan dipaparkan apabila anda klik pada butang *Current Report* atau butang *Passed Report* seperti yang dinyatakan dalam seksyen sebelumnya.

Laporan keseluruhan terdiri daripada 3 bahagian iaitu:

- Profil Analisa (*Analyzing Profiles*)
- Jadual Peratusan Plagiarisme (*Percentage of Plagiarism Table*)
- Rumusan (*Summary*)

Profil Analisa

Overall Report of Plagiarism

Analyzing Profiles

Subject : Komputeran Kembara(WRES3405)
Lecturer : Dr. Mazliza
Analyzing Date : Monday, 03-February-2003, 21:43:40
Analyzed Documents : 6 documents
Descriptions : Shows only 70% & above percentage of plagiarism
Go to summary...

Percentage of Plagiarism Tables

File 1	File 2	% File 1	% File 2	Severity	Details
WES000001.doc	WES000002.doc	100.00 %	100.00 %	High	View
WES000005.doc	WES000006.doc	82.84 %	90.98 %	High	View

Summary

It is recommended to you to check more further about the 4 potential documents with high severity of plagiarism as stated below :

1. WES000001.doc
2. WES000002.doc
3. WES000005.doc
4. WES000006.doc

High severity: 2 possibilities
Medium severity: 0 possibilities
Show the graph report...

[Back to top]

Jadual Peratusan Plagiarisme

Rumusan

Laporan Keseluruhan

Profil Analisa (*Analyzing Profiles*)

Plagiarizer 2003 memudahkan anda membuat rujukan ke atas laporan yang telah di jana dengan membekalkan seksyen Profil Analisa yang mana bahagian ini menunjukkan butiran maklumat mengenai :

- ✓ Nama subjek yang diuji serta kod subjek
- ✓ Nama pensyarah
- ✓ Tarikh dan masa set dokumen dianalisa
- ✓ Bilangan dokumen yang dianalisa
- ✓ Deskripsi mengenai aturan peratusan plagiarisme yang telah disetkan oleh anda

Jadual Peratusan Plagiarisme (*Percentage of Plagiarism Table*)

Seksyen ini menunjukkan jadual yang menunjukkan peratusan plagiarisme yang terlibat antara 2 fail dokumen yang mana jadual tersebut hanya akan menyenaraikan dokumen yang terlibat plagiarisme berdasarkan aturan peratus plagiarisme yang telah disetkan oleh anda.

Jadual tersebut terdiri daripada :

- ✓ Kolum **File 1** dan **File 2** masing-masing menunjukkan dokumen fail yang telah dibandingkan antara satu sama lain.
- ✓ Kolum **% File 1** dan **% File 2** merupakan peratusan plagiarisme antara fail 1 kepada fail 2 dan sebaliknya bagi kolum **% File 2**.
- ✓ Kolum **Severity** menunjukkan tahap keseriusan bagi kes plagiarisme yang berlaku antara 2 fail yang telah dianalisa dan dibandingkan.
- ✓ Kolum **Details** akan membenarkan anda klik pada pautan **View** untuk mendapatkan laporan yang terperinci.

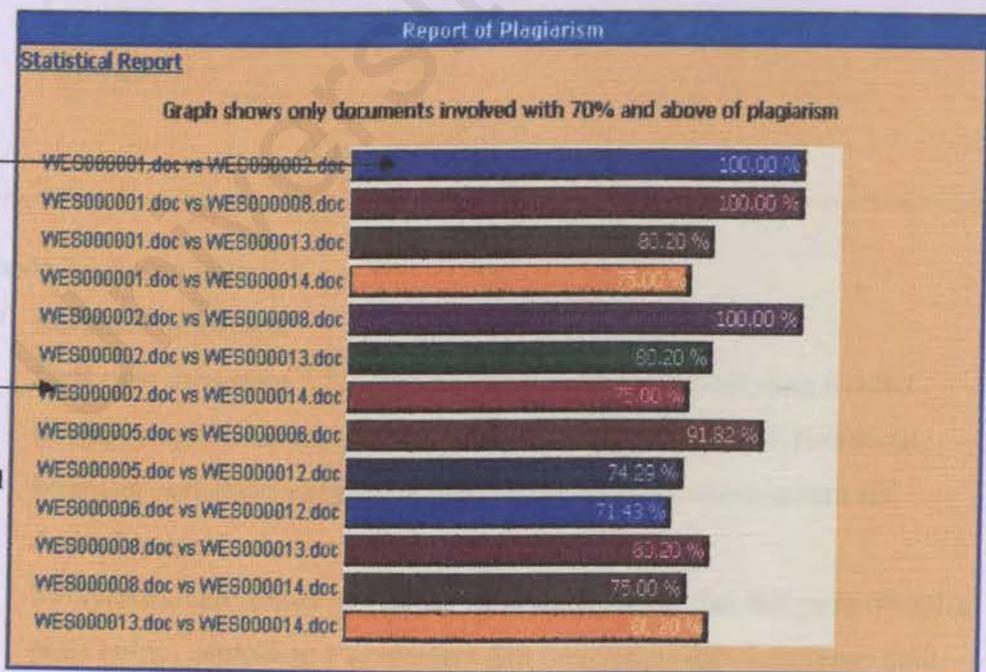
Rumusan(Summary)

Seksyen ini memudahkan anda untuk mengesan pelajar yang terlibat dengan plagiarisme yang mana ia akan :

- ✓ Menyenaraikan dokumen yang terlibat dengan plagiarisme dengan kadar 80% yang mana kadar ini merupakan tahap serius dan dinasihatkan agar anda membuat semakan lanjut ke atas dokumen tersebut.
- ✓ Menyenaraikan bilangan kebarangkalian yang terlibat dengan plagiarisme pada tahap serius dan tahap sederhana
- ✓ Menyediakan pautan *Show the graph report* yang mana ia akan memaparkan laporan yang berorientasikan graf.

2.7.3.2 Laporan Berorientasikan Graf

Apabila anda mengklik pada pautan *Show the graph report* yang terdapat di bahagian bawah laporan keseluruhan, laporan berorientasikan graf akan dipaparkan.



Graf peratusan

2 fail yang dibandingkan serta pautan ke laporan terperinci antara 2 fail tersebut

Laporan Berorientasikan Graf

2.7.3.3 Laporan Terperinci

Dipaparkan apabila anda klik pada :

- 1) Klik pada pautan **View** yang terdapat laporan keseluruhan **atau**;
- 2) Klik pada pautan nama 2 fail yang terdapat pada laporan berorientasikan graf

The screenshot displays a software interface for document analysis. On the left, there are two labels: 'Statistik Dokumen' with an arrow pointing to the 'Documents Statistic' section, and 'Jadual Terperinci' with an arrow pointing to the 'Details' section. The 'Documents Statistic' section shows the following information: File 1: WES000001.doc, File 2: WES000002.doc, Total Word Count File 1: 238 words, Total Word Count File 2: 238 words, Total matching: 238 words, Percentage matching file 1 against file 2: 100.00%, and Percentage matching file 2 against file 1: 100.00%. The 'Details' section contains a table with two columns: 'Matching Vocabulary' and 'Matching Frequency'. The table lists various words and their corresponding frequencies.

Matching Vocabulary	Matching Frequency
April	4
B	1
Begini	1
Dengan	4
Didapati	1
E	1
Feb	2
Jika	1
Julai	1
Kami	2
Kos	1
Maka	1
Penambahan	1

Laporan Terperinci

Terdiri daripada 2 seksyen iaitu :

✓ **Statistik dokumen**

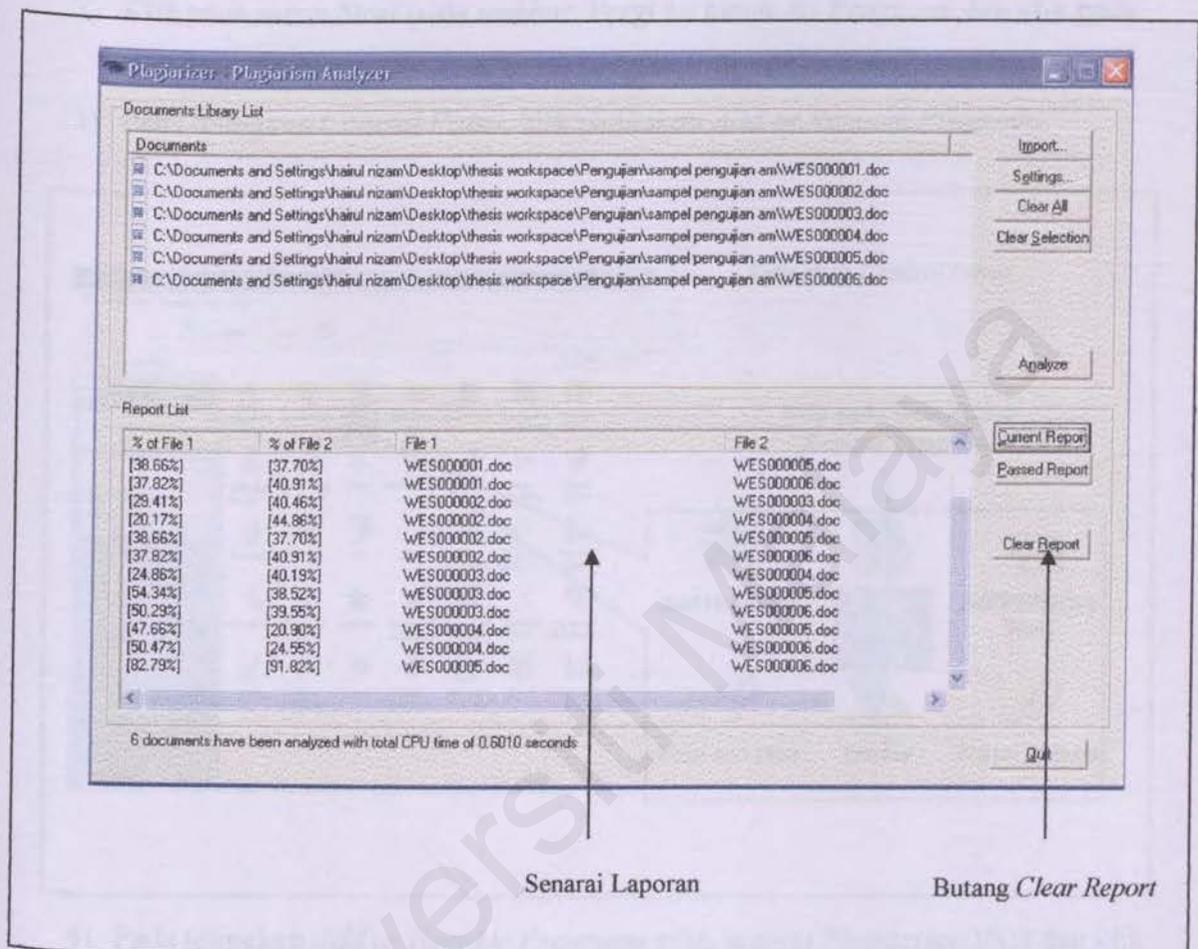
Deskripsi mengenai 2 fail dokumen yang dibandingkan yang terlibat dengan plagiarisme merangkumi nama dokumen, bilangan perkataan, bilangan perkataan yang sama dan peratusan plagiarisme antara fail

✓ **Jadual terperinci**

Menyenaraikan semua perkataan yang sama antara dua fail yang dianalisa pada kolom **Matching Vocabulary** dan menyenaraikan kekerapan atau frekuensi bagi perkataan tersebut pada kolom **Matching Frequency**

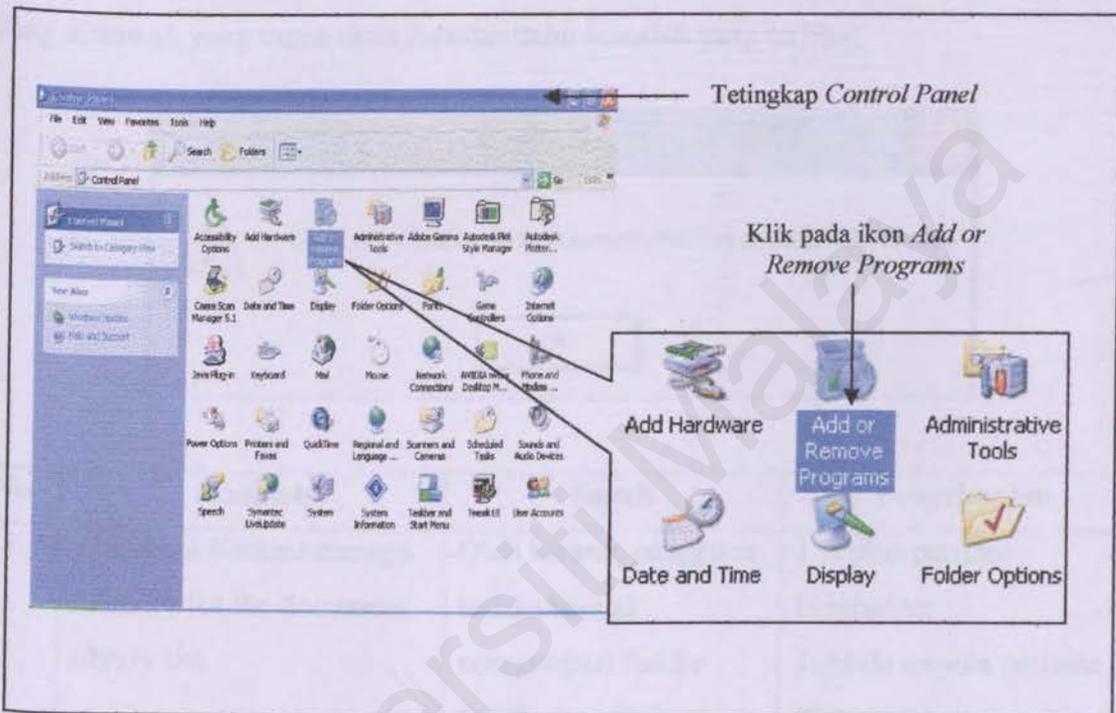
2.8 Mengosongkan Medan Senarai Laporan

1) Klik pada butang **Clear Report** untuk mengosongkan medan senarai laporan



3.0 Penyingkiran Perisian Plagiarizer 2003

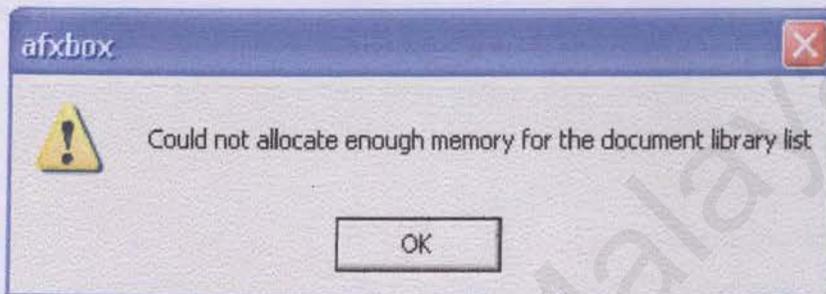
- 1) Keluar dan tutup aplikasi perisian Plagiarizer 2003 jika aktif
- 2) Klik pada menu *Start* pada *taskbar*. Pergi ke menu *All Programs* dan klik pada menu *Control Panel*
- 3) Pada tettingkap *Control Panel*, klik pada ikon *Add or Remove Programs*



- 4) Pada tettingkap *Add or Remove Programs* pilih senarai Plagiarizer 2003 dan klik pada butang *Change/Remove*
- 5) Satu tettingkap *pop-up Confirm File Deletion* dan klik butang *Yes* untuk menyingkirkan Plagiarizer daripada komputer anda atau butang *No* untuk membatalkan proses penyingkiran

4.0 Penyelesaian Masalah & Bantuan Pengguna

Seksyen ini akan menerangkan masalah-masalah yang mungkin timbul semasa menggunakan perisian Plagiarizer. Walaupun begitu, jika berlaku sebarang masalah ketika menggunakannya, perisian Plagiarizer akan mengeluarkan kotak amaran seperti yang di bawah yang mana akan memberitahu masalah yang terlibat.



No.	Masalah	Sebab	Penyelesaian
1.	Could not allocate enough memory for the document library list	Oleh kerana, pengguna terlalu banyak mengimport fail ke dalam perpustakaan perisian, komputer kekurangan sumber memori untuk menyimpan nama fail tersebut	1. Tutup perisian Plagiarizer 2. Mula semula perisian Plagiarizer 3. Import semula fail yang hendak diuji tetapi hadkan bilangan fail tersebut supaya tidak melebihi 30 dokumen per ujian
2.	Please import files to analyze	Pengguna tidak mengimport sebarang fail dokumen ke dalam perpustakaan perisian	1. Klik butang import 2. Import sekurang-kurangnya 2 fail dokumen untuk diuji

3.	Please import another files to analyze	Pengguna hanya mengimport satu fail dokumen ke dalam perpustakaan perisian	<ol style="list-style-type: none"> 1. Klik butang import 2. Import sekurang-kurangnya 2 fail dokumen untuk diuji
4.	Report folder "C:\Plagiarizer\Report" cannot be found	Folder laporan yang ditetapkan tidak dijumpai dalam komputer	<ol style="list-style-type: none"> 1. Cipta <i>folder</i> yang berkaitan dalam komputer anda dengan pergi ke <i>Windows Explorer</i>, klik kanan butang tetikus dan pilih menu <i>new</i> kemudian <i>folder</i> dan namakan <i>folder</i> tersebut dengan nama yang tertera dalam kotak amaran atau; 2. Klik butang <i>settings</i> kemudian butang <i>browse</i> dan pilih <i>folder</i> laporan yang sedia ada dalam komputer anda <p><i>*Tip – Perisian Plagiarizer secara default membuat set c:\plagiarizer\report\ sebagai folder untuk laporan</i></p>
5.	Cannot open document "C:\assignment\wres3405\wek000349"	Sumber dokumen tersebut mungkin telah diubah kedudukannya atau telah disingkirkan	<ol style="list-style-type: none"> 1. Import semula fail dokumen yang berkaitan dengan klik pada butang import

		sebelum perisian melakukan analisa	2. Pilih fail dokumen yang berkaitan
6.	Please select folder for report	Tiada sebarang <i>folder</i> laporan diset terlebih dahulu	1. Klik butang <i>settings</i> kemudian butang <i>browse</i> dan pilih <i>folder</i> laporan yang sedia ada dalam komputer anda

Jika wujud sebarang masalah selain daripada yang di atas, sila e-mel masalah anda ke semangat_daulat@perdana.um.edu.my atau semangat_daulat@hotmail.com

Universiti Malaysia