

SISTEM CAPAIAN MAKLUMAT BAGI KES PERUNDANGAN

Perpustakaan SKTM

**NORZITA BINTI DURIAT
WET 990151**

**LATIHAN ILMIAH YANG DIKEMUKAKAN UNTUK MEMENUHI
SEBAHAGIAN DARIPADA SYARAT MEMPEROLEHI
IJAZAH SARJANA MUDA TEKNOLOGI MAKLUMAT
DENGAN KEPUJIAN DALAM BIDANG
MULTIMEDIA**

JABATAN SISTEM DAN TEKNOLOGI KOMPUTER
FAKULTI SAINS KOMPUTER DAN TEKNOLOGI MAKLUMAT
UNIVERSITI MALAYA
KUALA LUMPUR

SESI 2002/2003

ABSTRAK

Laporan ini mengandungi isi kandungan lengkap berkenaan satu sistem yang akan saya bangunkan dengan nama **Sistem Capaian Maklumat Bagi Kes Perundangan**. Sistem yang akan saya bangunkan adalah bertujuan untuk membantu kakitangan firma guaman mengurus dan mencapai maklumat pelanggan dan kes-kes. Sistem ini akan menumpukan tentang maklumat berkenaan Undang-undang Kekeluargaan(*Family Law*) sahaja dan sistem ini diharapkan akan dapat menjadi satu contoh sistem yang mampu diaplikasikan ke dalam sistem perundangan yang lebih besar. Secara amnya, sistem ini akan mempunyai tiga modul iaitu Modul Carian, Modul Paparan dan Modul Input Maklumat. Ketiga-tiga modul ini akan mampu menjadikan sistem sebagai satu sistem komputer berpengkalan data yang mampu mengurus, mentadbir dan menyimpan data mengenai pelanggan dan juga kes-kes. Sistem ini juga diharapkan akan menjadi satu sistem yang lebih baik, cekap dan tepat bagi menggantikan sistem manual yang terbukti banyak kelemahannya.

PENGHARGAAN

Dengan nama Allah yang Maha Pemurah lagi Maha Penyayang, Alhamdulillah bersyukur di atas limpah rahmat dan keredhaanNya, maka dapatlah saya menyempurnakan Laporan Latihan Ilmiah I ini dalam masa yang ditetapkan.

Terlebih dahulu saya ingin mengucapkan ribuan terima kasih buat Cik Norisma Idris selaku penyelia projek di atas nasihat, cadangan dan sokongan yang diberi sepanjang perjalanan projek ini. Penghargaan ini juga di tujuarkan kepada Cik Nor Azlina Khamis selaku moderator di atas cadangan dan teguran yang telah diberikan.

Tidak ketinggalan juga kepada semua kakitangan dan staf Firma C.K Leong & Nurbaya yang telah memberi kerjasama yang sangat baik semasa sessi pencarian maklumat yang telah saya jalankan. Terima kasih di atas kesudian dan masa yang telah diberikan sehingga terhasilnya laporan ini.

Tidak lupa juga teristimewa buat keluarga tersayang yang banyak memberi dorongan serta rakan-rakan seperjuangan yang sentiasa memberi bantuan dari segi idea dan semangat sehingga terhasilnya laporan projek ini.

Terima Kasih Daun Keladi.

SENARAI ISI KANDUNGAN

ABSTRAK	ii
PENGHARGAAN	iii
SENARAI ISI KANDUNGAN	iv
SENARAI JADUAL	vii
SENARAI RAJAH	viii
BAB 1 : PENGENALAN	1
1.1 Latar Belakang Projek	1
1.2 Definisi Masalah	2
1.3 Objektif Projek	4
1.4 Skop Projek	6
1.5 Rancangan Pelaksanaan Projek	8
1.6 Strategi Pembangunan	9
1.7 Rumusan Projek	10
BAB 2 : KAJIAN LITERASI	12
2.1 Apakah Sistem Maklumat?	12
2.2 Sistem Pengurusan Pengkalan Data	15
2.3 Teknik Pencarian Maklumat	18
2.4 Analisis Maklumat	20
2.5 Sistem Yang Sedia Ada (<i>Analisis Dan Komen</i>)	23
BAB 3 : METODOLOGI	26
3.1 Model Pembangunan Yang Digunakan	26
3.2 Penerangan Setiap Fasa	28
3.3 Kajian Kemungkinan Sistem	33
BAB 4 : ANALISA SISTEM	35
4.1 Keperluan Fungsian	35
4.2 Keperluan Bukan Fungsian	38

4.3	Keperluan Perkakasan	39
4.4	Keperluan Perisian	40
BAB 5 : REKABENTUK SISTEM		48
5.1	Model Sistem	48
5.2	Carta Struktur	49
5.3	Gambarajah Konteks	49
5.4	Gambarajah Aliran Data	50
5.5	Carta Alir Setiap Modul Sistem	52
5.6	Kamus Data Sistem	55
5.7	Cadangan Antaramuka Sistem	56
5.8	Rekabentuk Input Output	57
BAB 6 : PERLAKSANAAN SISTEM		59
6.1	Pengenalan	59
6.2	Implementasi	60
6.3	Pembangunan	65
6.4	Fungsi-Fungsi Utama Sistem Capaian Maklumat Bagi Kes Perundangan	66
BAB 7 : PENGUJIAN SISTEM		69
7.1	Peringkat Pengujian	70
7.2	Proses Pengujian	72
7.3	Jenis-Jenis Ralat Dan Kesalahan	75
BAB 8 : PERBINCANGAN		78
8.1	Penilaian Sistem	78
8.2	Kelebihan Sistem	78
8.3	Kekangan Dan Penyelesaian	81
8.4	Cadangan Perluasan Sistem	84
8.5	Kesimpulan Perbincangan	85

SARANAN KESELURUHAN SISTEM	86
KESIMPULAN KESELURUHAN SISTEM	87

LAMPIRAN

PENGKODAN

RUJUKAN

MANUAL PENGGUNA

2.1 Manual Penggunaan Maklumat Perancangan Sistem Manual

29

SENARAI JADUAL

BAB 1 : PENGENALAN	9
BAB 1 : PENGENALAN	
1.1 Jadual Perancangan Projek	8
BAB 2 : KAJIAN LITERASI	13
BAB 2 : KAJIAN LITERASI	
2.1 Masalah Pengurusan Maklumat Perundangan Secara Manual	20
2.4 Jadual Pengurusan Maklumat Perundangan Secara Manual	21
2.5 Ciri-ciri Pengurusan Maklumat Secara Perkakpaper	22
BAB 3 : METODOLOGI	27
BAB 3 : METODOLOGI	
3.1 Model Ajar Terpadu	27
BAB 4 : PERAKERNAKAN SISTEM	38
BAB 4 : PERAKERNAKAN SISTEM	
4.1 Stuktur Sistem Operasi Maklumat Perundangan	38
4.2 Cara Struktur Hutan	40
4.3 Gambangjati Kecamatan	40
4.4 Gambangjati Kelurahan Selangor dan Pahang	40
4.5 Huj Kes Jenis	41
4.6 Cara Aliran Maklumat	42
4.7 Cara Aliran Maklumat	43
4.8 Maklumat Maklumat	43
4.9 Maklumat Maklumat Utama	45
4.10 Maklumat Maklumat Cetakan	46
4.11 Maklumat Maklumat Internet	47
BAB 5 : PENEMUAN SISTEM	51
BAB 5 : PENEMUAN SISTEM	
5.1 Maklumat Maklumat Yang Tidak Diketahui	51
BAB 6 : PENGEMBANGAN SISTEM	64
BAB 6 : PENGEMBANGAN SISTEM	
6.1 Cara Aliran Maklumat Pengetahuan	64

SENARAI RAJAH

BAB 1 : PENGENALAN

1.1	Strategi Pembangunan Sistem	9
-----	-----------------------------	---

BAB 2 : KAJIAN LITERASI

2.1	Komponen-komponen Sistem Maklumat	13
2.2	Menukar Data Kepada Maklumat	14
2.3	Komponen Sistem Pengkalan Data	16
2.4	Masalah Pengurusan Maklumat Perundangan Secara Manual	21
2.5	Sokongan Pengurusan Maklumat Secara Berkomputer	22

BAB 3 : METODOLOGI

1.1	Model Air Terjun	27
-----	------------------	----

BAB 5 : REKABENTUK SISTEM

5.1	Struktur Sistem Capaian Maklumat Bagi Kes Perundangan	48
5.2	Carta Struktur Sistem	49
5.3	Gambarajah Konteks	49
5.4	Gambarajah Aliran Data Bagi Sistem Capaian Maklumat Bagi Kes Perundangan	50
5.5	Carta Alir Modul Carian	52
5.6	Carta Alir Modul Input	53
5.7	Carta Alir Modul Paparan	54
5.8	Antaramuka Halaman Utama	56
5.9	Antaramuka Halaman Carian Data	56
5.10	Antaramuka Halaman Input	57

BAB 6 : PERLAKSANAAN SISTEM

6.1	Carta Struktur Sistem Yang Telah Diubah	64
-----	---	----

BAB 7 : PENGUJIAN SISTEM

7.1	Carta Aliran Maklumat Pengujian	70
-----	---------------------------------	----

BAB 1

PENGENALAN

BAB 1 : PENGENALAN

1.1 LATAR BELAKANG PROJEK

Sistem Capaian Maklumat Bagi Kes Perundangan ini adalah satu sistem berpengkalan data yang mampu menyimpan pelbagai maklumat pengguna dan kes perundangan. Secara amnya, sistem ini mampu melaksanakan 3 fungsi utama iaitu;

- a. Memasukkan data dan maklumat mengenai pelanggan firma dan juga apa-apa maklumat berkaitan perundangan.
- b. Memaparkan kembali maklumat yang telah dimasukkan.
- c. Mempunyai fungsi carian bagi mendapatkan maklumat pelanggan firma dan kes-kes perundangan.

Sistem ini saya bangunkan berdasarkan kepada pengumpulan maklumat dan perbandingan sistem manual yang telah digunakan oleh satu firma guaman yang telah saya pilih iaitu **Firma C.K Leong & Nurbaya** yang mana saya lihat mampu menjadi sumber maklumat dan domain sistem saya. Sistem ini tertumpu hanya kepada **Undang-Undang Kekeluargaan** sahaja, di mana setiap maklumat pelanggan dan kes-kes adalah berkenaan dengan **Undang-Undang Kekeluargaan**. Sistem ini hanyalah satu contoh bagi pembangunan sistem perundangan yang lebih besar.

Ia dibangunkan khas untuk kakitangan firma guaman meliputi kerani dan peguam. Sistem ini berkemungkinan menggunakan empat atau lima buah komputer sahaja dan akan menggunakan konsep rangkaian setempat atau Intranet yang hanya boleh dicapai di dalam rangkaian firma itu sahaja. Ini adalah kerana, maklumat pelanggan dan kes-kes itu sulit dan tidak boleh didedahkan dengan sewenang-

wenangnya. Oleh itu, sistem dilengkapi dengan kawalan capaian maklumat yang sangat ketat dengan adanya kata laluan dan kata khas pengguna. Sistem ini dibangunkan untuk menggantikan kaedah manual yang digunakan selama ini di mana ianya mempunyai banyak kelemahan dan kekurangannya.

Sistem ini mampu menyimpan maklumat pengguna di dalam satu pengkalan data dan memudahkan capaian kembali maklumat dengan lebih cepat dan sistematik. Fungsi carian(*search engine*) pula membolehkan kakitangan firma mencapai maklumat kes-kes perundangan terdahulu sebagai rujukan dan panduan bagi menyelesaikan kes-kes perundangan yang baru. Oleh itu, ia lebih kepada sistem yang membantu kakitangan firma guaman khususnya peguam untuk menyimpan maklumat yang berkaitan dengan pelanggannya dan melakukan rujukan kes-kes perundangan. Ini dapat mengelakkan kehilangan maklumat, kekeliruan dan pertindihan maklumat. Peguam boleh mencapai kembali maklumat, mengubah, menambah, mengemaskini bahkan menghapus maklumat-maklumat yang tidak dikehendaki.

1.2 DEFINISI MASALAH

Kebanyakan firma-firma guaman yang terdapat di negara kita tidak mempunyai satu sistem khas yang dibangunkan untuk membantu meringankan beban tugas para peguam dan kerani firma. Sebelum ini, para peguam hanya menggunakan sistem manual bagi penyimpanan maklumat-maklumat penting pelanggan dan kes-kes perundangan. Sistem manual yang di maksudkan ialah dengan menggunakan

sistem fail, disket dan “compact disc”. Penyimpanan maklumat dengan menggunakan sistem manual ini banyak kelemahannya. Antaranya ialah;

1.2.1. Capaian maklumat

Sukar untuk para peguam mencapai kembali maklumat pelanggan dan kes-kes perundangan yang dikehendaki. Di samping membuang masa dan tenaga, ianya juga memaksimakan kesilapan maklumat dan mendatangkan masalah dalam menangani sesuatu kes.

1.2.2. Kerosakan maklumat

Maklumat yang terlalu lama disimpan di dalam disket, fail atau “compact disc” akan mengalami kerosakan dan mengakibatkan kesilapan maklumat.

1.2.3. Kawalan maklumat

Sistem manual tidak dapat mengatasi kebocoran maklumat akibat daripada sistem manual yang tidak efisyen. Kebocoran ini boleh mendatangkan masalah sekiranya maklumat sampai kepada pihak-pihak yang tidak bertanggungjawab.

1.2.4. Pembaziran

Sistem manual banyak mendatangkan pembaziran wang ringgit, masa dan tenaga. Sistem manual memerlukan peruntukan kewangan untuk alatan penyimpanan seperti

kertas, fail, disket dan sebagainya. Di samping itu, mengakibatkan pembaziran masa dan tenaga dalam proses capaian kembali maklumat.

1.3 OBJEKTIF PROJEK

Sistem ini dibangunkan adalah untuk memenuhi beberapa objektif projek yang telah dikenalpasti seperti;

1.3.1. Penyelenggaraan maklumat

Pengumpulan maklumat dapat dilakukan dengan tepat, kemas dan teratur ke dalam sistem yang lebih efisyen dan sistematik.

1.3.2. Penyingkatan masa pemprosesan data

Memandangkan teknologi semakin maju, penjimatan masa menjadi faktor utama dalam produktiviti sebuah organisasi. Penggunaan sistem dapat mengelakkan masa pelanggan dan pihak firma guaman dibazirkan. Malah masa yang dijimatkan dapat digunakan untuk melaksanakan tugas lain (*David M.Kroenke 1995*).

1.3.3. Kemudahan capaian data

Maklumat dan akta-akta perundangan adalah banyak dan memerlukan pengurusan data yang cekap dan terkini. Ini adalah untuk memastikan kelancaran perlaksanaan tugas. Berdasarkan kenyataan ini maka dua

keadaan yang diberi perhatian iaitu bagaimana untuk mengendalikan kerja-kerja penstoran dan bagaimana untuk mencapai semula apabila dikehendaki. Faktor-faktor ini yang memberi keperluan untuk mewujudkan sebuah pangkalan data yang mampu menguruskan data secara on-line dan off-line bersesuaian dengan masa sekarang (*David M.Kroenke 1995*).

1.3.4. Kekonsistenan data

Data yang terkini dan konsisten adalah penting dalam pengurusan pangkalan data. Pentadbiran data yang tidak efisyen akan menyebabkan kerugian dan reputasi yang kurang baik terhadap sesebuah organisasi.

1.3.5. Pengurangan kos

Pengambilalihan kerja-kerja manual oleh komputer lebih menjimatkan kos dalam jangka masa panjang. Malah ia lebih konsisten dan keupayaannya dapat ditentukan memandangkan komputer dapat ditentukan prestasinya. Justeru itu perancangan tugas dan produktiviti akan lebih dipertingkatkan.

1.3.6. Keselamatan data

Penggunaan kertas boleh menyebabkan meningkatkan kebocoran data kepada pihak yang tidak bertanggungjawab. Tetapi dengan adanya sistem keselamatan seperti ini dapat mengelakkan dan mengurangkan

paras kebocoran data melalui penggunaan kata laluan capaian yang terhad (*David M.Kroenke 1995*).

1.3.7. Analisis data

Analisis data secara manual akan mendorong berlakunya kesilapan. Kelebihan yang ada sekarang ialah teknologi pemprosesan data ke dalam bentuk yang boleh difahami dan mudah diinterpretasikan.

1.3.8. Meminimumkan kesilapan

Dengan menggunakan sistem ini kesilapan dalam pengrekodan dapat dikurangkan ke tahap minimum. Berbanding sistem manual, sering berlaku kesilapan dalam merekod data, pengiraan dan pertindihan data.

1.4 SKOP PROJEK

1.4.1. Skop sistem

1.4.1.1 Proses Memasukkan Data Pengguna

Proses membenarkan peguam memasukkan segala data berkaitan kes yang sedang dikendalikan oleh pengguna. Meliputi jenis kes, data peribadi anak guam, akta-akta yang berkaitan dengan dan juga apa-apa pindaan terkini kepada akta-akta tersebut (*Latest Amendment Of The Acts*).

1.4.1.2 Proses Paparan Semula Data Pengguna

Membenarkan capaian kembali data bagi tujuan rujukan, pengemaskinian, pindaan, pembetulan atau penghapusan data.

1.4.1.3 Enjin carian

Membenarkan pengguna melakukan fungsi carian bagi kes-kes perundangan yang berkaitan dan juga maklumat pelanggan. Carian adalah melibatkan keseluruhan pengkalan data.

1.4.2 Skop Pengguna

Sistem dibangunkan khas untuk kemudahan peguam dan kakitangan firma guaman untuk memudahkan bebanan tugas mereka.

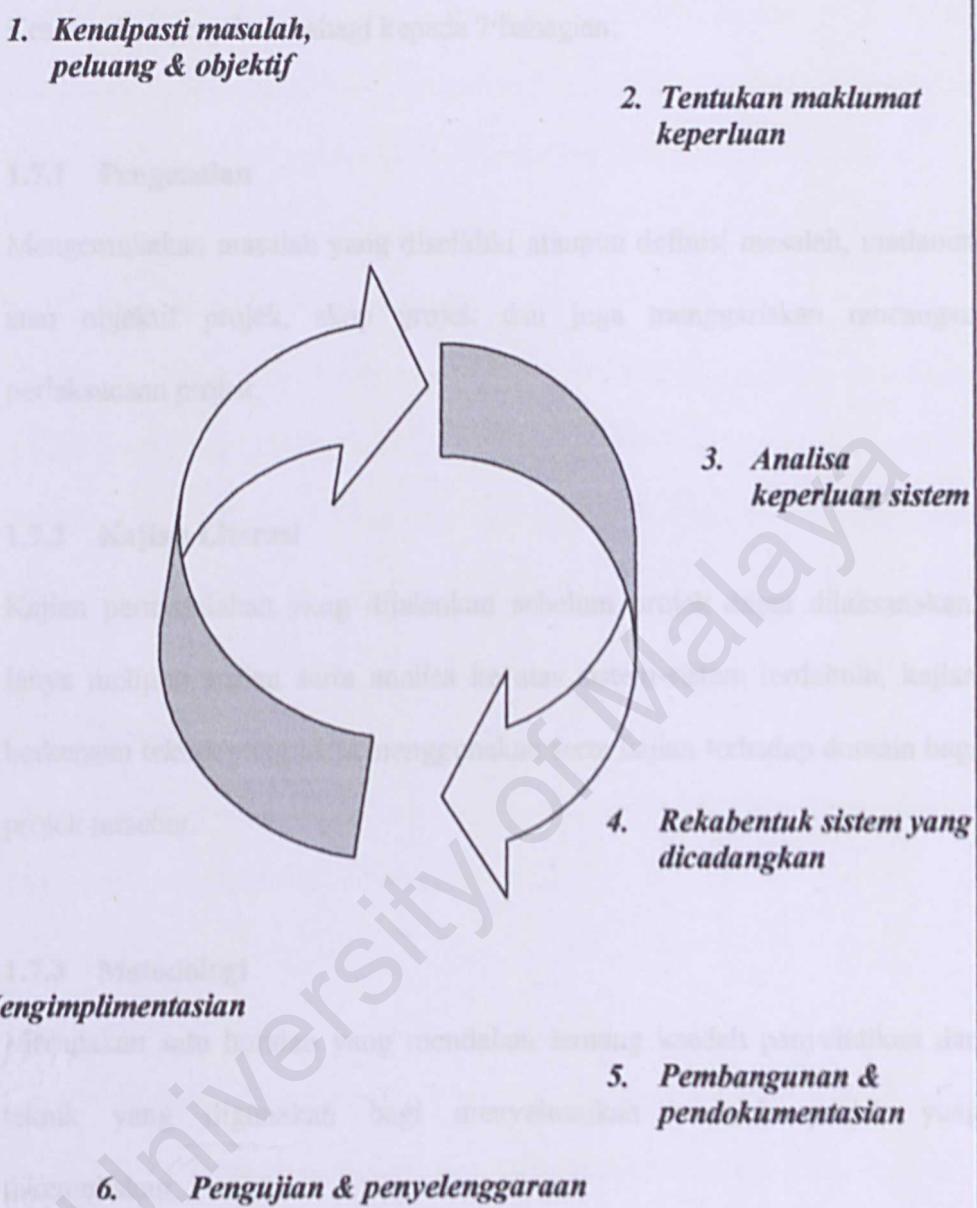
1.5 RANCANGAN PELAKSANAAN PROJEK

Pembangunan projek terbahagi kepada 7 fasa utama, dan rancangan pelaksaan projek dapat ditunjukkan dalam Jadual 1.1 di bawah.

Jadual 1.1 : Jadual Perancangan Projek

Bil	Aktiviti	Tarikh									
		Ju 2002	Aug 2002	Sep 2002	Oct 2002	Nov 2002	Dec 2002	Jan 2003	Feb 2003	Mar 2003	Apr 2003
1	Penggeralan										
2	Kajian Literasi										
3	Metodologi										
4	Analisa										
5	Rekabentuk										
6	Pembangunan										
7	Pengujian										
8	Perundingan										

1.6 STRATEGI PEMBANGUNAN



Rajah 1.1 : Strategi Pembangunan Sistem

1.7 RUMUSAN PROJEK

Keseluruhan projek terbahagi kepada 7 bahagian;

1.7.1 Pengenalan

Mengemukakan masalah yang diselidiki ataupun definisi masalah, matlamat atau objektif projek, skop projek dan juga menggariskan rancangan perlaksanaan projek.

1.7.2 Kajian Literasi

Kajian permasalahan yang dijalankan sebelum projek dapat dilaksanakan. Ianya meliputi kajian serta analisa ke atas sistem-sistem terdahulu, kajian berkenaan teknik yang akan menggunakan serta kajian terhadap domain bagi projek tersebut.

1.7.3 Metodologi

Merupakan satu huraian yang mendalam tentang kaedah penyelidikan dan teknik yang digunakan bagi menyelesaikan masalah projek yang dikemukakan.

1.7.4 Analisa Sistem

Huraian berkenaan keperluan-keperluan yang diperlukan seperti keperluan fungsian, keperluan bukan fungsian, keperluan perkakasan dan perisian.

1.7.5 Rekabentuk Sistem

Huraian yang melibatkan proses pencantuman kesemua bahagiaan tertentu kepada sebuah sistem yang emngandungi fungsi-fungsi yang harus dilaksanakan oleh sistem. Ianya melibatkan rekabentuk skrin atau antaramuka, aliran maklumat dan modul-modul yang terlibat dalam sesuatu projek.

1.7.6 Perlaksanna / Pembangunan Sistem

Huraian berkenaan pembangunan sistem yang merujuk kepada penukaran modul-modul dan algorithma yang telah direkabentuk ke dalam arahan-arahan yang boleh dilaksanakan menggunakan bahasa pengaturcaraan komputer yang tertentu.

1.7.7 Pengujian Sistem

Huraian yang mengesahkan sama ada sistem berfungsi mengikut keperluan dan spesifikasi yang telah ditentukan.

1.7.8 Perbincangan

Membincangkan keputusan yang diperolehi, masalah dan penyelesaian, kelebihan dan kelemahan sistem yang dibangunkan, peningkatan yang boleh dijalankan pada masa hadapan, cadangan serta kesimpulan projek yang dijalankan.

BAB 2

KAJIAN LITERASI

BAB 2 : KAJIAN LITERASI

Sebelum membangunkan sesuatu sistem atau memulakan sesuatu projek, kajian dan literasi perlu dilakukan. Ini adalah untuk mendapatkan maklumat dan fakta penting bagi membantu pembangunan menghasilkan satu sistem yang baik dan memenuhi kehendak pengguna serta ramah pengguna iaitu untuk dapatkan hasil akhir yang memuaskan. Kajian literasi di sini adalah lebih kepada penerangan secara terpeinci tentang domain projek saya dan hal-hal yang berkaitan dengan domain.

2.1 APAKAH SISTEM MAKLUMAT?

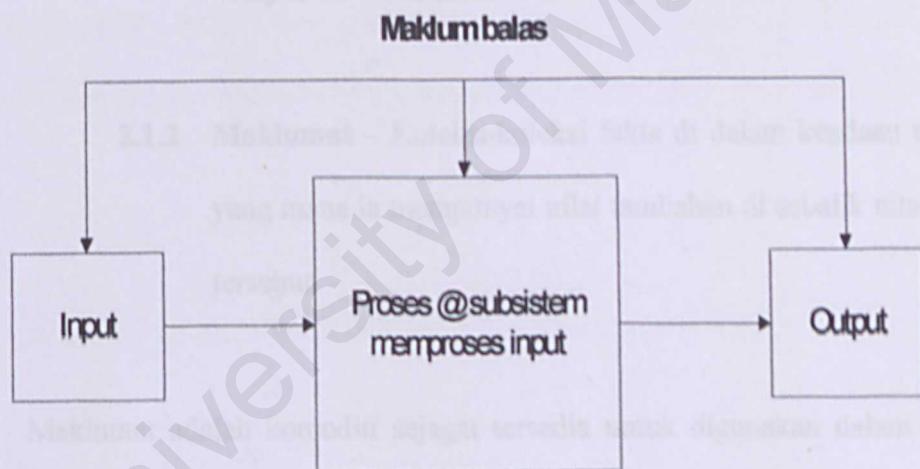
Menunjuk kepada gabungan awalan kerja, maklumat, individu dan teknologi maklumat. Ia diatur secara tatasusunan dengan tujuan untuk mencapai matlamat organisasi.

Dalam sistem ini, terdapat beberapa gabungan yang membolehkan ia berfungsi dengan lebih berkesan dan tersusun. Amalan kerja adalah kaedah yang digunakan oleh individu dan teknologi maklumat dalam melakukan kerja. Ini adalah berteraskan kepada sistem manual yang digabungkan dan dikemaskini dengan teknologi maklumat untuk menghasilkan mutu kerja yang baik.

Maklumat merujuk kepada data yang telah diformat, teks, gambar, imej dan bunyi sementara **Teknologi Maklumat** merangkumi perkakasan dan perisian yang

digunakan untuk melaksanakan tugas-tugas pemprosesan seperti pemindahan, penyimpanan, capaian, manipulasi dan pemaparan data.

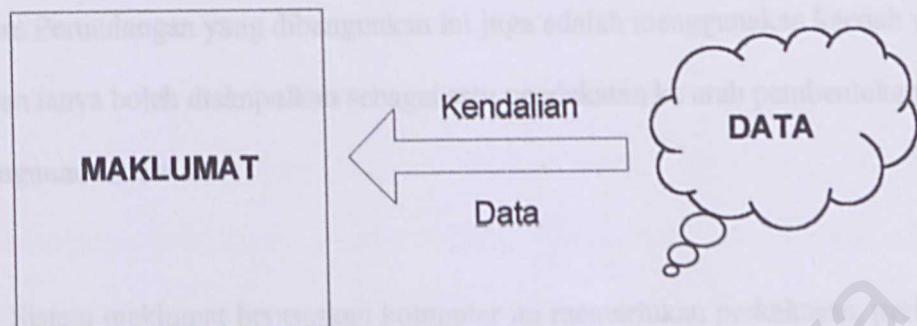
Sistem Maklumat merupakan satu sistem yang terkini di mana ia merupakan sistem yang melibatkan input kepada sistem. Bagi menghasilkan satu operasi yang berupa maklumat, sistem ini perlu melalui beberapa proses atau subsistem yang lain kerana gabungan yang melalui proses sahaja mampu menghasilkan output tersebut. Input daripada sistem merupakan data yang penting bagi sistem seperti memasukkan data pengguna, penyimpanan data dan sebagainya. Komponen-komponen sistem maklumat dapat dilihat dalam **Rajah 2.1** di bawah.



Rajah 2.1 : Komponen-komponen Sistem Maklumat

Terdapat 2 elemen penting untuk menjadikan sistem maklumat teratur dan terurus. Gabungan kedua-duanya akan menjadikan sistem maklumat yang mampu beroperasi secara sistematik tanpa keperluan manual lain. Elemen-elemen tersebut ialah;

2.1.1 Data – Terdiri daripada fakta-fakta mentah. Daripada data, maklumat akan mengambil data bagi menuarkannya ke dalam satu bentuk yang lebih bermakna. Proses penukaran tersebut dapat dilihat pada Rajah 2.2 di bawah.



Fakta yang diproses mempunyai bentuk dan corak

Rajah 2.2 : Menukar Data Kepada Maklumat

2.1.2 Maklumat – Koleksi-koleksi fakta di dalam keadaan terurus yang mana ia mempunyai nilai tambahan di sebalik nilai fakta tersebut.

Maklumat adalah komoditi sangat tersedia untuk digunakan dalam semua bentuk maklumat. Ianya menambah nilai hanya bila berjumpa dengan keperluan sebenar dan apabila diperlukan. Ia akan diuruskan dan disebarluaskan dalam keadaan dan bentuk yang sistematik. Bagi menghasilkan sesuatu maklumat itu bernilai, ia perlu menepati beberapa ciri penting seperti ia mesti tepat, lengkap dan ekonomik untuk dihasilkan. Ia bertujuan mengelakkan sesuatu maklumat yang diterima itu palsu atau tidak menepati istilah sebenar maklumat dengan keadaan yang dihasilkan adalah tidak terurus dan teratur, anjal, boleh dipercayai, berkaitan, mudah difahami dan tepat pada masanya. Jelaslah kepada kita bahawa 2 elemen penting yang

memerlukan antara satu sama lain itu akan merangkumi sebuah maklumat yang membawa maksud yang besar. Kesempurnaan sistem maklumat adalah datangnya dari 2 elemen tersebut dan apabila ia bergabung, akan terbentuklah satu maklumat yang menjadi sebahagian daripada sistem itu berfungsi. Sistem Capaian Maklumat Bagi Kes Perundangan yang dibangunkan ini juga adalah menggunakan kaedah yang sama dan ianya boleh disimpulkan sebagai satu pendekatan ke arah pembentukan dan pembangunan sistem ini.

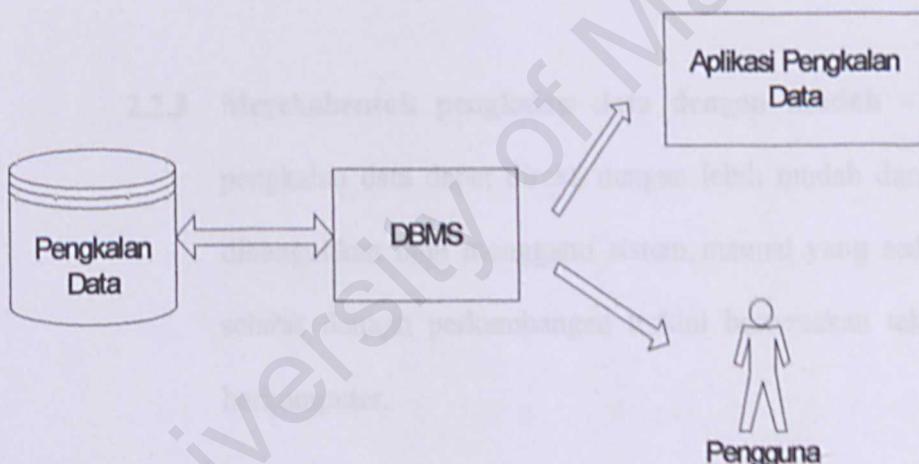
Sistem maklumat berdasarkan komputer ini memerlukan perkakasan, perisian, pengkalan data, telekomunikasi, prosedur dan manusia bagi mencapai sasarannya. Dibangunkan sebagai satu sistem yang mampu digunakan dalam apa jua pengurusanselaras dengan kemajuanyang menekankan aspek penkomputeran, serta mengantikan sistem manual bagi meningkatkan lagi sistem pengoperasian.

2.2 SISTEM PENGURUSAN PENGKALAN DATA

Pengkalan data membawa makna sebagai koleksi data yang berkongsi dan dapat digunakan untuk pelbagai tujuan oleh pelbagai pengguna. Ianya juga dapat dilihat bahawa pengkalan data dapat menjurangkan ulangan data untuk berkongsi data di antara beberapa pengguna. Dapat difahami bahawa dengan menggunakan pengkalan data ini, ianya dapat membantu para pengguna menyediakan sesuatu maklumat itu dengan lebih tepat dan sempurna tanpa perlu mengulang semula faktafakta yang sedia ada bagi tujuan penggubahan. Ianya juga boleh dikesan sebagai satu cara yang mampu mewujudkan satu rangkaian pengurusan yang lebih teratur

dengan adanya pengkalan data di mana pelbagai jenis maklumat dapat dikongsi bersama antara pengguna.

Sesebuah komputer seharusnya mempunyai beberapa ciri penting yang sama yang disebut Sistem Pengurusan Pengkalan Data (*DBMS*) yang merupakan satu sistem perisian yang membolehkan pengguna menakrif, mencipta, mengemaskini, menyusun, mengurus dan menyelenggara pengkalan data. DBMS juga mengawal capaian terhadap data dalam pengkalan data. Pendeknya, DBMS memudahkan pengurusan data dan maklumat yang diperlukan oleh sesebuah organisasi. DBMS bertindak sebagai perantara di antara pengguna dan pengkalan data. Ini dapat digambarkan seperti dalam Rajah 2.3.



Rajah 2.3 : Komponen Sistem Pengkalan Data

Kebanyakan sistem pengurusan pengkalan data ini menyokong bahasa yang sedia ada yang digunakan untuk menulis program dengan lebih mudah. Antara kebaikan sistem pengurusan pengkalan data ialah;

2.2.1 Capaian sesuatu data – maklumat diperolehi dengan lebih cepat dan mudah. Terbukti ia lebih sempurna daripada sistem

pengurusan manual kerana capaian data yang mudah, ringkas, cepat dan teratur.

2.2.2 Mengurangkan pembaziran – dalam bentuk masa, wangs dan pembaziran yang melibatkan data kerana sistem pengurusan pengkalan data ini dapat menambah dan memansuh maklumat yang sedia ada tanpa perlu ubah data atau program. Dengan ini dapat memudahkan sebuah pengurusan beroperasi dan hanya satu rangkaian sahaja diperlukan untuk menghubungkan satu bahagian dengan bahagian yang lain.

2.2.3 Merekabentuk pengkalan data dengan mudah – suatu pengkalan data dapat direka dengan lebih mudah dan dapat dibangunkan bagi mengganti sistem manual yang sedia ada selaras dengan perkembangan terkini berteraskan teknologi berkomputer.

Sistem pengurusan pengkalan data yang baik bergantung kepada cara operasiannya dan bagaimana sistem pengurusan itu digunakan. Seharusnya sistem dapat mengurangkan jumlah masa yang diperlukan untuk mengatur, menyimpan dan iaanya dapat mengembalikan maklumat. Secara amnya sistem pengurusan pengkalan data ini sepatutnya menyediakan kemudahan-kemudahan tertentu bagi memudahkan pengguna. Antaranya;

- i. Mencipta pengkalan data
- ii. Menambah data ke dalam pengkalan data
- iii. Menukar atau membaiki data yang sedia ada
- iv. Menghapus data yang sedia ada di dalam pengkalan data
- v. Menyisih rekod-rekod yang ada
- vi. Capaian kembali data untuk pertanyaan atau penjanaan laporan dari pengkalan data

2.3 TEKNIK PENCARIAN MAKLUMAT

Saya telah menggunakan 5 teknik dalam proses pencarian maklumat;

2.3.1 Bahan Pencetak – banyak idea dan konsep yang berkaitan boleh diperolehi daripada buku, majalah dan akhbar.

2.3.2 Laman Web Di Internet – dengan melayari laman-laman web yang berkaitan dengan perundangan di internet telah memberikan saya idea tentang rekabentuk sistem dan bagaimana ia beroperasi. Antara laman web yang saya lawati ialah;

2.4.2.1 <http://www.courtreporter.law.com/>

2.4.2.2 <http://www.findlaw.com/>

2.4.2.3 <http://www.iplcca.com/>

2.4.2.4 <http://www.lawnet.com/>

Laman-laman web yang disenaraikan tadi adalah laman-laman web berkaitan dengan perundangan di dalam Malaysia dan di luar negara.

2.3.3 Perbincangan – saya telah melakukan perbincangan dan temubual secara formal dan tidak formal dengan peguam-pegawai dan kakitangan firma **C.K Leong & Nurbaya** bagi mendapatkan maklumat dan hal-hal yang berkaitan dengan perundangan. Saya telah menyediakan set soalan temubual bagi mendapatkan maklumat dan ternyata teknik ini banyak membantu saya kerana dapat berhubung terus dengan peguam dan bertanya masalah berkenaan sistem manual yang mereka gunakan selama ini.

2.3.4 Sumber Laporan Projek-Projek Sebelum Ini – bahan-bahan cadangan projek di dalam laporan projek terdahulu telah memberikan maklumat tentang sumber, cara pengolahan dan susunan bahasa juga bagaimana cara untuk menulis laporan dan membangunkan sistem yang baik.

2.3.5 Mengedarkan Soalan Kaji Selidik – dengan mengedarkan set soalan mengenai sistem manual dan maklumat-maklumat perundangan kepada kakitangan firma **C.K Leong & Nurbaya**.

2.4 ANALISIS MAKLUMAT

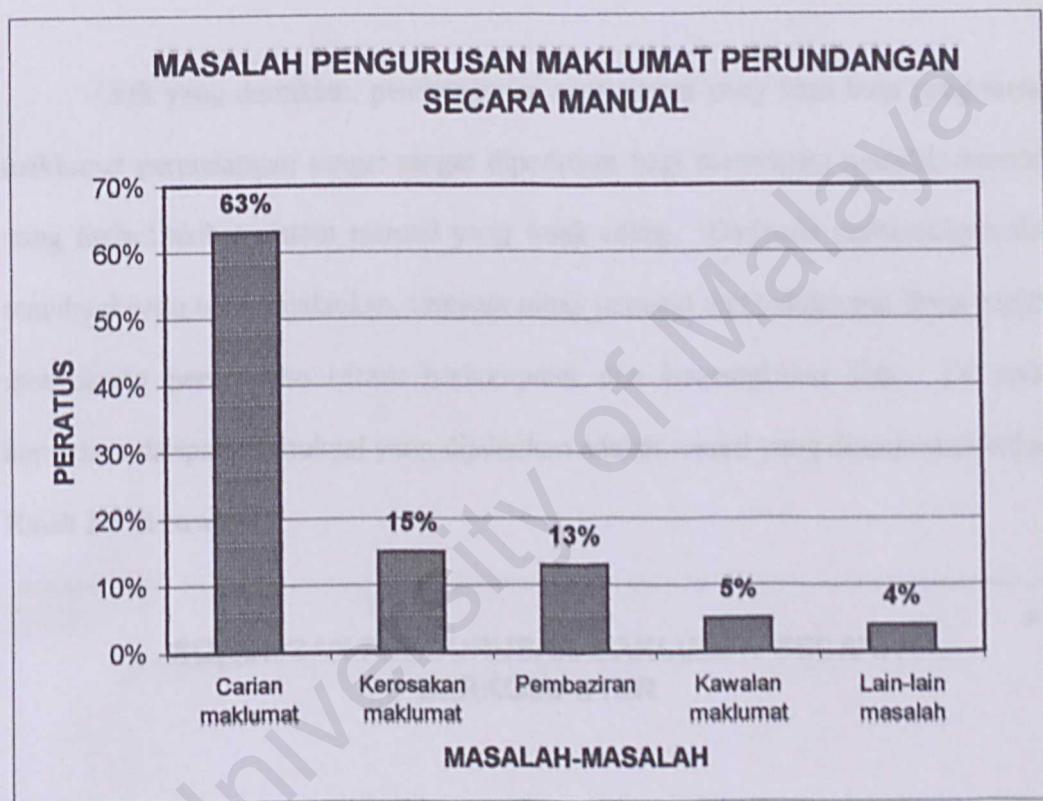
Daripada proses pencarian maklumat, didapati bahawa pelbagai masalah telah dikenalpasti menjadi punca kelemahan sistem pengurusan maklumat secara manual. Masalah-masalah tersebut telah menunjukkan bahawa pengurusan maklumat secara manual banyak kelebihannya. Masalah-masalah yang dikenalpasti tersebut bukan sahaja mengganggu prestasi kerja kakitangan sesebuah organisasi, malah akan memburukkan imej baik firma di kalangan pelanggan.

Antara masalah yang paling mendatangkan kesulitan kepada kakitangan khususnya peguam adalah masalah capaian maklumat diikuti dengan kerosakan maklumat, pembaziran dan juga kawalan keselamatan maklumat. Jadual 2.1 di bawah menunjukkan masalah-masalah yang dihadapi dan tahap peratusannya.

Jadual 2.1 : Masalah Pengurusan Maklumat Perundangan Secara Manual

BIL	MASALAH	PERATUS(%)
1	Carian maklumat	63%
2	Kerosakan maklumat	15%
3	Pembaziran	13%
4	Kawalan maklumat	5%
5	Lain-lain masalah	4%

Daripada jadual, didapati bahawa masalah yang melibatkan capaian maklumat adalah yang paling tinggi dengan 63%, diikuti dengan kerosakan maklumat 15%, pembaziran 13%, kawalan maklumat 5% dan lain-lain masalah 4%. Bagi menunjukkan peratus masalah yang lebih jelas lagi, di bawah disertakan Rajah 2.4 untuk mewakili masalah-masalah di atas dengan lebih jelas dengan menggunakan graf garis.



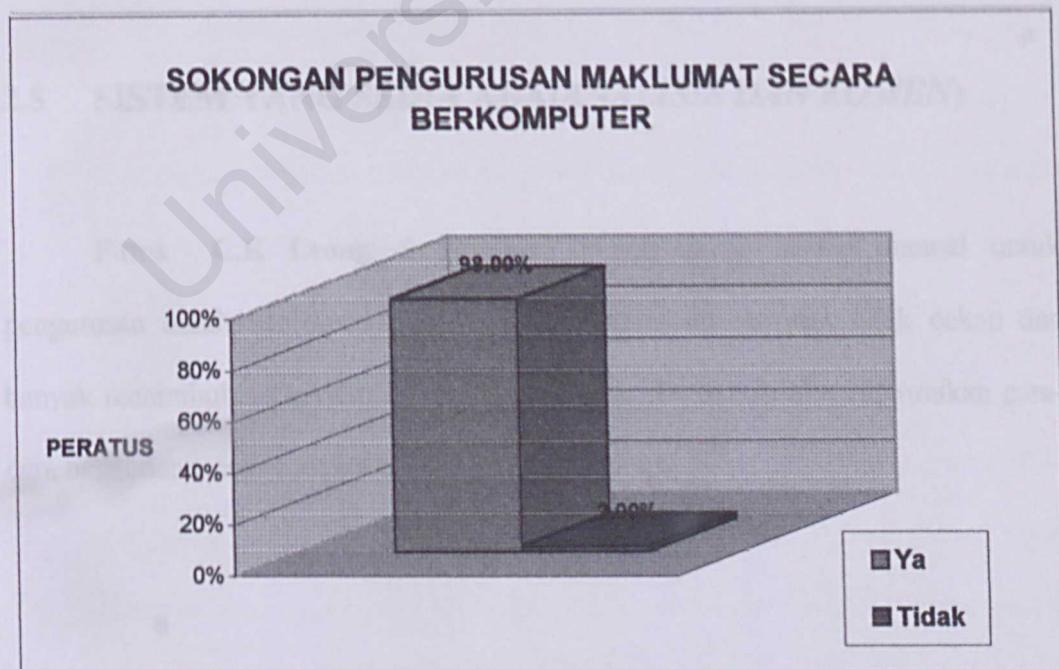
Rajah 2.4 : Masalah Pengurusan Maklumat Perundangan Secara Manual

Daripada masalah yang paling tinggi iaitu carian maklumat, telah didapati bahawa terdapat beberapa kategori carian yang menyukarkan pengguna secara manual iaitu;

- Capaian bagi kes-kes perundangan yang berkaitan.
- Capaian bagi maklumat pelanggan.

Ini adalah kerana terdapat beratus-ratus kes-kes perundangan yang telah diselesaikan sebelum ini dan capaian maklumat kes-kes tersebut secara manual adalah menyukarkan pengguna. Kerana kemungkinan besar ia telah hilang atau rosak disebabkan oleh faktor penyimpanan yang tidak efisyen dan sistematik. Maklumat pelanggan juga banyak dan pihak firma wajib menyimpan maklumat sehingga 6 bulan. Maka semestinya memerlukan satu tempat penyimpanan maklumat yang besar dan teratur.

Oleh yang demikian, pembangunan satu sistem yang khas bagi pengurusan maklumat perundangan sangat-sangat diperlukan bagi menangani masalah-masalah yang timbul akibat sistem manual yang tidak cekap. Daripada perbincangan dan temubual yang telah dijalankan, ternyata pihak peguam dan kakitangan firma sangat menyokong pengurusan secara berkomputer dan berpengkalan data. Di mana keputusan daripada temubual yang dijalankan adalah seperti yang ditunjukkan dalam Rajah 2.5 di bawah.



Rajah 2.5 : Sokongan Pengurusan Maklumat Secara Berkomputer

Daripada Rajah 2.5, didapati bahawa 98% daripada pengguna menyokong pengurusan secara berkomputer manakala 2% lagi tidak bersetuju. Dengan ini, dapat di buat kesimpulan bahawa pengguna sangat memerlukan satu sistem berkomputer yang boleh memudahkan penyelenggaraan maklumat sekaligus memudahkan tugas dan bebanan kerja mereka.

• Cakera kerja (compact disk)

Namun begitu, kebanyakkan kakitangan dan peguam syarikat C.K Leong & Nurbaya adalah terdiri daripada mereka yang kurang biasa dengan teknologi dan jarang sekali menggunakan satu sistem khas yang mengendalikan pengurusan secara berkomputer. Oleh itu, rekabentuk sistem haruslah mudah, teratur, "ramah pengguna" dan bebas ralat. Rekabentuk antaramuka pengguna adalah yang paling penting bagi membantu pengguna menggunakan sistem dengan lebih baik. Antaramuka pengguna yang baik dapat mengurangkan kadar masa mempelajari sistem dan memudahkan pengendalian sistem.

2.5 SISTEM YANG SEDIA ADA(ANALISIS DAN KOMEN)

Firma C.K Leong & Nurbaya menggunakan sistem manual untuk pengurusan maklumat mereka. Sistem tradisional ini ternyata tidak cekap dan banyak menimbulkan kesulitan kepada pengguna. Di bawah akan diuraikan cara-cara bagaimana sistem secara tradisional ini berfungsi.

2.5.1 Cara Penyimpanan Maklumat

Cara penyimpanan maklumat adalah secara manual sepenuhnya.

Segala maklumat akan disimpan dengan 3 cara iaitu;

- Fail dan jilid
- Disket
- Cakera keras (compact disc)
- Ingatan komputer

Ketiga-tiga cara penyimpanan tersebut akan dilakukan sekiranya terdapat maklumat baru tentang pelanggan atau maklumat berkaitan dengan kes. Setiap fail, disket dan cakera keras akan di jilidkan atau diberikan nombor rujukan bagi memudahkan pengguna mencarinya kelak. Setelah itu, akan disimpan di dalam satu bilik maklumat khas yang telah disediakan oleh pihak pentadbir.

Setelah menilai sistem simpanan maklumat yang digunakan, nyata sekali firma tersebut memerlukan satu sistem berkomputer yang mampu menangani masalah penyimpanan maklumat yang tidak cekap. Setiap maklumat mampu ditakbir dengan baik dan cekap.

2.5.2 Cara Capaian Maklumat

Capaian maklumat yang digunakan adalah secara tradisional. Pengguna perlu mencari sendiri fail, disket atau cakera keras yang mengandungi maklumat di dalam bilik maklumat dengan berpandukan kepada nombor rujukan yang telah diberikan.

Carian maklumat dengan cara ini sangat menyusahkan dan membuang masa dan tenaga. Tempoh masa yang lama diperlukan untuk mencari maklumat yang dikehendaki. Sedangkan dengan sistem berkomputer, proses carian dapat dilakukan dengan singkat.

BAB 3

METODOLOGI

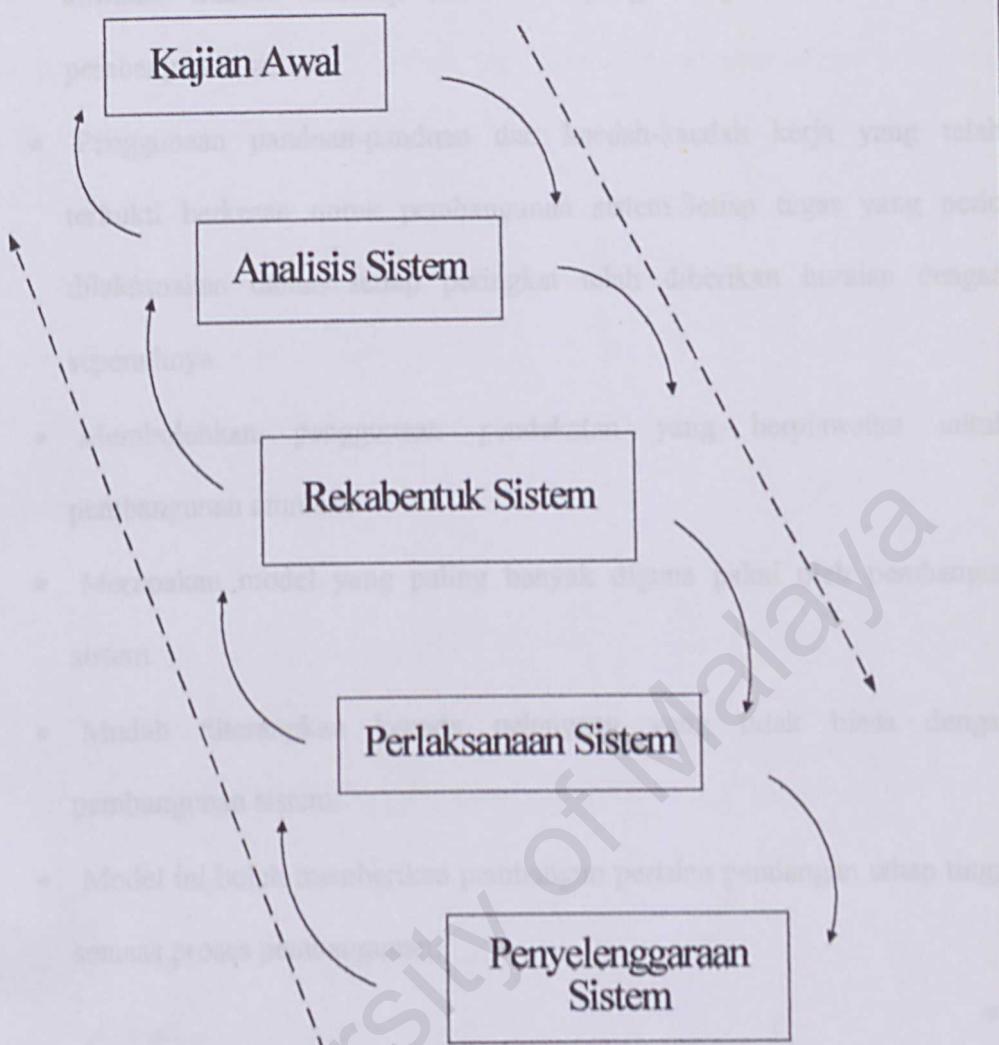
BAB 3: METODOLOGI

Model metodologi proses pembangunan perlu ditentukan terlebih dahulu sebelum memulakan sesuatu projek pembangunan sistem. Ini adalah untuk memastikan setiap perancangan projek dapat dilaksanakan dengan baik dan menepati pelan perancangan seperti yang telah digariskan. Metodologi proses pembangunan adalah penting untuk membantu pembangun sistem untuk mewujudkan pemahaman yang sama terhadap aktiviti, sumber dan kekangan juga mengenalpasti ketidak konsistenan, lewahan dan perkara-perkara lain yang tertinggal di dalam proses juga fasa-fasa yang terlibat serta dapat memastikan apa yang dirancang dapat dilaksanakan.

3.1 MODEL PEMBANGUNAN YANG DIGUNAKAN

Model pembangunan sistem yang dipilih adalah **Model Air Terjun (Waterfall)**. Model pembangunan sistem ini mengandungi bidang-bidang kerja yang tersusun, bermula dengan peringkat kajian permulaan hingga ke peringkat sistem tersebut dilaksanakan dan seterusnya diselanggarakan.

Pemilihan langkah-langkah pembangunan yang teratur bertujuan untuk memastikan bahawa tujuan pembangunan sistem dan tujuan setiap bidang kerja oleh mereka yang terlibat dalam pembangunan sistem. Metodologi ini telah terbukti berkesan dan diterima pakai sehingga kini. Rajah 3.1 di bawah menunjukkan Peringkat Fasa Model Air Terjun.



Rajah 3.1 : Model Air Terjun

Antara sebab mengapa metodologi ini dipilih ialah;

- Memberikan panduan pembangunan sistem yang berstruktur dan sistematik daripada fasa ke fasa yang lain.
- Mempunyai ciri kekitaran iaitu proses pengulangan boleh dijalankan jika sebarang perubahan berlaku.

- Perlaksanaan projek dapat dikawal dengan sempurna dan membolehkan kawalan kualiti terhadap hasil-hasil yang dicapai daripada projek pembangunan sistem.
- Penggunaan panduan-panduan dan kaedah-kaedah kerja yang telah terbukti berkesan untuk pembangunan sistem. Setiap tugas yang perlu dilaksanakan dalam setiap peringkat telah diberikan huraihan dengan sepenuhnya.
- Membolehkan penggunaan pendekatan yang berpiawaian untuk pembangunan aturcara.
- Merupakan model yang paling banyak diguna pakai oleh pembangun sistem.
- Mudah diterangkan kepada pelanggan yang tidak biasa dengan pembangunan sistem.
- Model ini boleh memberikan pembangun perisian pandangan tahap tinggi semasa proses pembangunan.

3.2 PENERANGAN SETIAP FASA

Bahagian ini akan menerangkan fungsi dan aktiviti setiap fasa pembangunan keseluruhan sistem.

3.2.1 Kajian Awal

Fasa ini juga dikenali juga sebagai fasa tafsiran masalah yang merupakan peringkat pertama dalam usaha membangunkan sistem. Semasa perlaksanaan fasa

ini, pendapat-pendapat tentang penyelesaian masalah mungkin akan dikemukakan oleh pihak pengguna. Dengan melakukan kajian awal tentang masalah dan keperluan sistem semasa yang wujud, maka mudah bagi mencapai matlamat seperti kenalpasti skop dan kelebihan projek.

Aktiviti-aktiviti yang dilakukan semasa fasa Kajian Awal ialah;

- Memahami masalah semasa
- Memahami apakah projek yang hendak dilaksanakan
- Menentukan objektif sistem
- Menentukan skop sistem
- Menyediakan skedul projek
- Memilih dan menentukan model
- Pembangunan sistem untuk perlaksanaan
- Modul pembangunan sistem
- Mengenal pasti sasaran pengguna

Daripada fasa 1 ini, telah didapati bahawa sistem semasa yang wujud di dalam proses pengurusan maklumat bagi perundangan adalah secara manual. Sistem yang dijalankan pada hari ini tidak begitu cekap dan sistematik. Ini menimbulkan beberapa masalah antaranya;

- Memerlukan satu sistem baru yang lebih mudah dan cepat iaitu sistem berkomputer.
- Kesukaran untuk menyimpan dan mencapai data-data yang telah lama kerana kemungkinan tempat untuk menyimpan rekod telah hilang atau rosak.

- Kemungkinan data hilang semasa membuat pertukaran tempat atau pengemaskinian rekod-rekod lama.

Oleh itu, satu sistem baru iaitu sistem berkomputer diperlukan. Sistem ini dapat mengurangkan penggantungan kepada kertas dan tempat penyimpanan rekod-rekod pelanggan dan kes perundangan di mana rekod-rekod ini boleh disimpan di dalam satu tempat yang sama dan mudah untuk dicapai pada bila-bila masa. Maklumat ini juga boleh dipaparkan apabila dikehendaki dan mempunyai pelbagai kelebihan seperti;

- Dapat menjamin kekonsistenan data – kemaskinian maklumat boleh dilaksanakan serentak di pelbagai pihak di samping menjamin tiadanya tindanan maklumat.
- Capaian maklumat dapat dilakukan dalam jangkamasa singkat.
- Risiko kehilangan data dapat dikurangkan.
- Pemprosesan maklumat dapat diperingkatkan dari segi kelajuan dan kelancaran.
- Maklumat-maklumat yang penting dan sulit dapat dilindungi daripada pengguna-pengguna tidak sah.
- Mengurangkan kos dalam penggunaan kertas dan lain-lain bahan penyimpanan manual.

3.2.2 Analisis Sistem

Fasa ini melibatkan beberapa keperluan bagi sesebuah sistem yang akan dibangunkan seperti;:

- Menentukan keperluan sistem

- Menganalisis keperluan sistem
- Mendokumentasikan keperluan sistem
- Membuat keputusan

Untuk fasa ini maklumat daripada fasa 1 adalah penting untuk mengalisisnya dengan teliti. Dalam fasa ini juga, aspek-aspek utama bagi menghasilkan sistem berdasarkan web ini telah dianalisa, dikaji dan dinilai.

3.2.3 Rekabentuk Sistem

Fasa ini bermatlamat bagi menghasilkan satu sistem yang berkesan dan bersesuaian untuk digunakan oleh sasaran pengguna. Ia juga boleh dipercayai dan boleh dicapai dengan berkesan serta mudah diselanggarakan, fleksibel serta mudah dimodifikasi. Tujuan utama ialah supaya menghasilkan satu sistem yang dinamik dan interaktif.

Aktiviti yang telibat dalam fasa ini ialah;

- Mengkaji kehendak dan keperluan
- Merekabentuk skrin paparan
- Merekabentuk imej
- Merekabentuk animasi
- Merekabentuk borang
- Merekabentuk pangkalan data.

Dengan mengkaji dan meneliti dokumentasi keperluan sistem, proses pembangunan seterusnya akan menjadi lebih mudah. Rekabentuk sistem akan mengambil kira kesesuaian dengan sasaran pengguna, maklumat yang

hendak disampaikan, tahap keramahan pengguna dan kemampuan menarik pengguna. Kesemua aktiviti-aktiviti yang disenaraikan diatas akan menghasilkan prosedur model sistem.

3.2.4 Pelaksanaan Sistem

Proses menukar rekabentuk yang telah ditetapkan kepada implementasi sebenar dengan menyediakan kod sumber dan kod program bagi membentuk sistem. Dalam proses ini, pengaturcara akan sentiasa belajar dari kesilapan pengaturcaraan, buku-buku dan aspek konfigurasi sistem. Dalam proses ini juga model air terjun telah dipilih untuk modul pembangunan sistem. Ia dipilih kerana bersesuaian dengan sistem yang akan dibangunkan serta berada dalam jadual yang ditetapkan.

Peringkat fasa secara terperinci;

- Pembinaan Pangkalan data
- Peringkat pengaturcaraan
- Peringkat ujian
- Peringkat penerimaan
- Peringkat perlaksanaan dan operasi

Dapatlah disimpulkan bahawa fasa ini bertujuan untuk menghasilkan perancangan yang lebih sistematik berkenaan proses penjadualan, pengkodan dan pengujian. Pada fasa ini juga pembangunan sistem telah siap sepenuhnya dan sedia untuk digunakan oleh pengguna dalam keadaan sebenar. Setelah itu penilaian akan dibuat terhadap sistem samada mengikut kehendak pengguna atau tidak.

3.2.5 Penyelenggaraan Sistem

Dalam fasa ini, pengujian akan dijalankan sepanjang masa. Modul-modul sistem yang dipilih akan diuji bagi menentukan kesilapan dan kesesuaian samada memenuhi matlamat sistem. Penyelenggaraan akan dilakukan bagi mengubah mana-mana kesilapan yang dikesan sepanjang proses pengujian dilakukan.

Setelah fasa pembangunan sistem telah selesai sepenuhnya, maka kan dilakukan **Dokumentasi Sistem** di mana merangkumi proses-proses;

- Penyediaan laporan projek
- Penyediaan laporan pengguna
- Menyediakan format persembahan untuk menerangkan perjalanan dan hasil sistem kepada penyelia dan “moderator”

3.3 KAJIAN KEMUNGKINAN SISTEM

3.3.1 Kajian Kemungkinan Teknikal

Sistem ini boleh digunakan pada komputer peribadi dan pencetak yang sedia ada dengan adanya modem dan sambungan kepada rangkaian setempat(*intranet*).

3.3.2 Kajian Kemungkinan Operasi

Sistem ini sangat diperlukan setelah dikenalpasti banyak kelemahan dalam sistem manual. Ia juga dapat menjimatkan masa, tenaga disamping cepatkan proses penyimpanan dan pencarian.

3.3.3 Kajian Kemungkinan Ekonomi

Tidak memerlukan kos yang banyak untuk membeli perkakasan baru melainkan firma pengguna tidak mempunyai komputer, modem dan peralatan yang diperlukan.

BAB 4

ANALISA SISTEM

BAB 4 : ANALISA SISTEM

Keperluan adalah suatu ciri sistem atau penerangan tentang sesuatu yang boleh dilakukan oleh sistem bagi memenuhi tujuan sistem tersebut. Biasanya, keperluan dibahagi kepada 3 kategori;

- a. Keperluan yang perlu dipenuhi dan dicapai secara mutlak.
- b. Keperluan yang boleh diadakan tetapi tidak diperlukan.
- c. Keperluan yang mungkin diperlukan tetapi boleh ditiadakan jika tidak berkenaan.

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

4.1 KEPERLUAN FUNGSIAN

Sebagai satu sistem komputer, sistem ini mempunyai beberapa fungsi utama di mana setiap fungsi tersebut mempunyai peranan masing-masing. Antara fungsi utama yang akan terdapat di dalam sistem adalah;

4.1.1 Carian :

Fungsi ini adalah fungsi utama di mana pengguna biasanya hanya mahu melihat maklumat yang mereka kehendaki sahaja dan tidak dipaparkan dengan perkara-perkara yang dirasakan tidak perlu. Oleh itu, sistem akan terus meminta pengguna memasukkan carian mereka selepas sahaja mereka memasukkan login ID dan katalaluan yang betul. Mereka diberi pilihan

samaada untuk melakukan carian untuk **maklumat pelanggan** atau **kes-kes perundangan**.

Carian akan dilakukan berdasarkan kata kekunci yang dimasukkan oleh pengguna. Sebarang maklumat yang sama dengan kata kekunci yang di masukkan akan dipaparkan sebagai hasil carian.

4.1.2 Paparan :

Selepas sesuatu carian dilakukan, sistem akan memaparkan ringkasan soalan tersebut. Pengguna perlu memilih daripada hasil carian (“search result”) yang dipaparkan untuk pergi ke kes yang dicari. Hasil paparan (“search result”) dipaparkan dengan kata kunci dan sedikit petikan daripada maklumat yang dicari. Sebagai contoh, apabila pengguna memilih carian untuk **kes-kes perundangan** dan memasukkan kata kunci “*child custody*”, maka sistem akan memaparkan:

1. *Family Law – child custody- consent term for custody – whether can be amended- whether consent from each parties*
.....
2. *Family Law – child custody- whether right of custody for child below 7 years shall be vested in the mother as of right.....*

Untuk pengguna mendapatkan maklumat yang telah dicari, hanya perlu ‘klik’ pada ‘hypertext’ supaya laman yang mengandungi maklumat dipaparkan.

4.1.3 Keselamatan :

Untuk mengelakkan kemasukan pengguna tanpa kebenaran, peraturan kemasukan ke dalam sistem perlu dilaksanakan dengan berkesan. Sistem ini menggunakan fungsi login ID dan kata laluan yang hanya diketahui oleh pembangun sistem dan pengguna yang sah sahaja. Apabila pengguna memasuki sistem, mereka diminta memasukkan login ID dan kata laluan yang telah ditetapkan. Cubaan login sistem adalah sebanyak tiga(3) kali sahaja. Pengguna akan dihalang daripada memasuki sistem sekiranya login kali yang ke tiga(3) tidak berjaya. Satu mesej akan dipaparkan memberitahu yang pengguna telah dihalang daripada memasuki sistem dan dikehendaki menghubungi individu tertentu untuk memdapatkan ID dan kata laluan yang betul kepada pengguna sistem.

ID dan kata laluan sistem mestilah mempunyai ciri-ciri unik dan tersendiri supaya tidak dapat dikesan oleh pihak-pihak yang diingini.

4.1.4 Tambah :

Fungsi ini dicapai apabila pengguna hendak menambah maklumat yang telah disimpan.

4.1.5 Hapus :

Pengguna dikehendaki memilih maklumat yang ingin dihapuskan dan menghapuskannya satu demi satu.

4.1.6 Kemaskini :

Sekiranya pengguna menyedari kesilapan pada maklumat yang disimpan, atau berlaku pembetulan. Ini adalah untuk memastikan maklumat yang terdapat di dalam sistem sentiasa tepat dan tidak mengelirukan.

4.3 KEPERLUAN PERKARASAN

4.2 KEPERLUAN BUKAN FUNGSIAN

Merujuk kepada had-had atau kekangan terhadap perkhidmatan-perkhidmatan yang disediakan oleh sistem. Ia juga mengambil kira had masa semasa proses pembangunan sistem. Berikut adalah keperluan bukan fungsian sistem yang akan dibangunkan.

- Mesra pengguna – pengguna mudah faham dan belajar untuk menggunakan sistem.
- Membalas segera – proses pencarian dijangka akan lebih cepat di mana ia hanya perlu membuat pertanyaan pada kata kunci unik dan segala maklumat yang dikehendaki segera dipaparkan.
- Kebolehgunaan – pengguna mudah untuk menggunakan sistem ini samaada dari segi memasukkan, mengemaskinikan atau mencapai kembali data.
- Kebolehgunaan semula – modul-modul di dalam sistem mestilah boleh diguna semula sekiranya pengubahsuaian diperlukan pada masa hadapan. Setiap modul boleh digunakan secara berasingan.

- Kebolehpercayaan – mengeluarkan output seperti mana yang dikehendaki oleh pengguna bila digunakan di dalam keadaan normal. Ianya juga perlu dibina dengan pengesan ralat dan pemaparan mesej jika kesilapan berlaku.

4.3 KEPERLUAN PERKAKASAN

Perkakasan secara minimum adalah terbahagi kepada 2 bahagian iaitu keperluan semasa pembangunan sistem dan semasa masa larian sistem yang telah siap dibina.

4.3.1 Semasa Pembangunan Sistem

Perkakasan paling minimum.

- Komputer dengan pemproses minimum Pentium 166 MHz
- Sistem pengoperasian Windows 95 ke atas.
- Kapasiti cakera keras(*harddisc*) 4.3 GB
- Memori 64 MB SDRAM
- Monitor 15 inci
- Papan kekunci, tetikus

4.3.2 Semasa Larian Sistem

Perkakasan paling minimum

- Komputer dengan pemproses minimum Pentium 166 MHz
- Sistem pengoperasian Windows 95 ke atas.
- Kapasiti cakera keras(*harddisc*) 4.3 GB
- Memori 64 MB SDRAM

- Monitor 15 inci
- Papan kekunci, tetikus
- Pencetak
- Pemacu cakera liut 1.44 MB
- Modem

4.4 KEPERLUAN PERISIAN

Adalah merupakan pemilihan teknologi yang benar-benar sesuai memenuhi fungsi-fungsi atau item-item yang terdapat dalam sistem yang akan dibangunkan. Ini amat penting dalam memastikan pengguna mendapat faedah dari output yang maksima.

Perisian yang dipilih tidak semestinya terlalu hebat atau kompleks, sebaliknya ia dapat memberikan persembahan sistem yang baik dan memenuhi kepuasan pengguna sistem dari segi masa, tindak balas, keupayaan sokongan pelbagai dokumen serta berupaya untuk menarik perhatian pengguna.

4.4.1 Active Server Pages (ASP)

Merupakan suatu perisian yang baik untuk membangunkan laman web yang dinamik. Ia sebenarnya bukanlah bahasa pengaturcaraan, sebaliknya merupakan satu teknologi yang membenarkan arahan HTML diprogramkan hanya selepas penghantaran laman web tersebut dilarikan. Istilah teknologi adalah samar kerana ia tidak seperti bahasa pengaturcaraan

yang lain seperti C++ maupun Visual Basic, tetapi ia wujud dalam bahasa seperti Javascript dan VBScript. Ia juga merupakan aplikasi seperti FrontPage atau Word2000.

Kekuatan ASP adalah dalam 2 perkara;

1. ASP tidak direka sehingga pengguna ingin melihat laman web
2. ASP juga tidak sensitif terhadap web larian yang digunakan.

Ini kerana ia boleh dilaksanakan pada mana-mana komputer server yang menyokongnya.

ASP juga membenarkan pembangun sistem untuk membuat sebarang penambahan terhadap laman web dengan ciri-ciri interaktif. Ia juga sensitif terhadap faktor-faktor seperti masa dan tempat, pilihan pengguna sebelumnya dan tindakbalas.

Antara kebaikan ASP ialah;

a. **Pembangunan Aplikasi Yang Cepat**

- Pembangunan halaman web dapat dilakukan dengan menggunakan piawai model berorientasikan objek yang sama dalam persekitaran windows. Kita boleh menggunakan teknik berorientasikan objek yang piawai untuk membangunkan aplikasi secara web. Oleh kerana kita hanya perlu menulis skrip yang sedikit dan menggunakan skrip yang sama, kita boleh mencipta halaman dengan lebih cepat.
- Kebebasan pelayar dan platform

- Kita boleh menggunakan penskriptan model berorientasikan objek tanpa mengira jenis pelaya yang mencapai aplikasi tersebut. Secara maya, ia berfungsi tanpa berfungsi tanpa mengira samada penskriptan dilakukan secara pelayan mahu pun pelanggan.
- Kita boleh mencipta borang dengan lebih menarik kawalan “design-time” ke atas mukasurat seperti mana dilakukan pada persekitaran Visual Basic. Kita juga boleh mengasingkan logik aplikasi dalam prosedur diskret. Ini termasuklah dengan menghubungkait prosedur pada mukasurat lain dengan mukasurat asal.
- Penskriptan objek mewujudkan suatu mekanisma yang membenarkan kita melayari dari satu mukasurat ke mukasurat yang lain dengan menspesifikasikan nama daripada menspesifikasikan URL mukasurat tertentu.

b. Aspek-Aspek Keselamatan.

Dari aspek keselamatan pula, pengaturcara penskriptan ActiveX membenarkan sepenuhnya pengaturcara membina aturcara keselamatan sendiri tanpa apa-apa sekatan dan penghada. Maka keselamatan halaman adalah di tangan pengaturcara itu sendiri dan ia bebas menyediakan paras keselamatan mengikut keperluan sistem.

4.4.2 Microsoft Frontpage 2000

Merupakan salah satu perisian yang terbaik dalam menyunting web. Ia dibuat oleh Microsoft™ dan mempunyai ciri-ciri merekabentuk tapak web yang kompleks, interaktif dengan hanya beberapa klik dari tetikus. Menyediakan Personal Web Server yang boleh menguruskan tapak web dengan sendiri dan sebarang perubahan pada web boleh di “upload” secara terus. Tiada editor html lain yang dapat dapat membenarkan perekaan tapak web dengan automasi, kekonsistenan dan kesesuaian yang ditawarkan oleh Frontpage.

Perisian ini juga membenarkan ciri-ciri CGI skrip digunakan dengan menggunakan Frontpage WebBot untuk melaksanakan fungsi-fungsi yang lebih sukar seperti pemprosesan borang, perbincangan atas talian “online discussion” dan sebagainya.

Ia juga membenarkan penggunaan “template” yang sedia ada dan ini memudahkan pengguna tanpa perlu memikirkan apakah rekabentuk antaramuka yang sesuai bagi web mereka.

Frontpage membenarkan pengimportan sebarang dokumen pemprosesan perkataan dan fail .html. Pilihan tetingkap juga membenarkan pengubahsuaian sumber kod html secara atas talian “online”. Ia juga membenarkan sebarang pengimportan grafik imej dari sebarang format dan Frontpage secara automatik akan menuarkannya ke dalam bentuk .GIF atau .JPEG fail.

4.4.3 Microsoft Access 2000

Pengkalan data boleh dianggap sebagai satu set fail besar yang berkaitan secara logik, disusun untuk dipermudahkan capaian oleh satu atau lebih aturcara penggunaan dan untuk meminimumkan limpahan data. Konsep ini tidak bermakna bahawa semua data yang berkaitan mesti terkandung dalam pengkalan data tunggal tetapi semua rekod dalam semua pengkalan dat sepatutnya ada kaitan dan limpahan data perlu diminimumkan.

Perisian Microsoft Access 2000 telah dipilih sebagai pengkalan data sistem yang akan dibangunkan ini adalah kerana ia sesuai untuk dilaksanakan pada platform Microsoft Windows 95/98/ME/NT/2000 yang selalu digunakan bagi komputer peribadi dan ia merupakan salah satu aplikasi di dalam Microsoft 2000. perisian ini juga serasi dengan pengaturcaraan yang akan digunakan iaitu ASP, Javascript dan VBScript.

Perisian ini menyediakan peralatan yang membolehkan pengguna untuk menyimpan semua maklumat mengenai pengkalan data. Di samping itu, ia menyediakan peralatan bagi memudahkan proses kemaskini maklumat di dalam pengkalan data yang dilaksanakan.

Dengan menggunakan perisian ini, proses pengujian dapat dijalankan dengan lebih mudah tanpa perlu melakukan larian modul-modul yang tidak terlibat. Ia juga dapat menjimatkan masa pembangunan sistem dan proses pembetulan boleh dilakukan secara terus. Di samping itu, menawarkan pelbagai ciri baru untuk merekabentuk dan penggunaan pengkalan data yang

lebih mudah, ini termasuklah kemudahan untuk memahami penentuan kunci-kunci utama dan hubungan-hubungan antara jadual.

4.4.4 Adobe Photoshop 6.0

Adobe Photoshop adalah antara alatan penyunting imej yang akan digunakan, manakala animagic GIF animator akan digunakan bagi menghasilkan animasi. Ciri-ciri dan kemudahan yang ditawarkan oleh Adobe Photoshop ialah;

- Menyediakan kemudahan lapisan demi lapisan "layer by layer" bagi memudahkan proses penyuntingan tanpa mengubah lapisan lain. Proses memadam juga menjadi mudah dengan adanya kemudahan lapisan tersebut.
- Menyediakan butang-butang memodifikasi imej bagi penyediaan kesan imej yang banyak.
- Membenarkan penerimaan "plug-in" yang boleh memuat turun dari internet bagi membolehkan penambahan keupayaan atau ciri-cirinya.

4.4.5 Java Script

Java script ialah merupakan pengaturcaraan yang agak mudah. Ianya telah diciptakan oleh kumpulan Netscape Communicator Inc. ia boleh disesuaikan dengan semua pelayar Netscape 2.0 dan ke atas dengan lebih baik sekali. Ia adalah sebuah aturcara yang kecil di dalam laman web yang akan dilaksanakan oleh pengguna web. Masa dan ciri-ciri perlaksanaannya dikawal oleh pembangun yang menulis skrip. Fungsinya ialah ia boleh

dipanggil di dalam satu dokumen web dan ia biasanya dilaksanakan oleh fungsi tetikus, bebutang atau papan kekunci. Ia juga boleh digunakan untuk mengawal sepenuhnya pelayar Netscape dan Microsoft Internet Explorer. JavaScript tidak sama dengan Java yang merupakan bahasa pengaturcaraan berasingan. Pernyataan JavaScript lebih mudah dan ringkas serta bahasa pengaturcaraan yang berintrepetasi. Sebaliknya, Java adalah suatu bahasa pengaturcaraan berorientasikan objek yang terkompil.

JavaScript berciri dinamik di mana ia boleh ditakrifkan semula pada bila-bila masa. Ia seterusnya akan memberikan kemudahan dan fleksibiliti yang lebih baik dan amat berguna dalam pengaturcaraan. Di samping itu, ia berupaya dalam menilai kod-kod yang ada sepanjang masa larian. Ini akan membolehkan pengaturcara menakrifkan semula secara dinamik penggantungan logik semasa larian. JazaScript bukan bahasa berorientasikan objek seperti Java, sebaliknya merupakan alternatif yang berkesan berdasarkan prototaip.

Di antara kelebihan JavaScript ialah;

- Ia dijamin dapat dilarikan menggunakan mana-mana pelayar.
- Ia boleh ditakrifkan semula pada bila-bila masa. Ini memberikan lebih fleksibiliti kepada skripnya.
- Ia mampu menilai aturcara semasa larian bagi membolehkan pengaturcara untuk membuat takrifan terhadap kepergantungan logik aturcara semasa ia dilarikan.

- Ia amat berguna kepada pelayan “server” kerana berupaya untuk membuat pencarian menerusi rentetan ungkapan.
- JavaScript sebenarnya bukan sahaja dapat digunakan untuk sistem pengoperasian Windows tetapi juga untuk lain-lain pelantar “platform”. JavaScript merupakan bahasa pengaturcaraan yang tidak bergantung kepada pelantar yang digunakan, oleh itu aplikasi yang dibangun menggunakannya boleh dijalankan pada semua sistem pengoperasian.

BAB 5

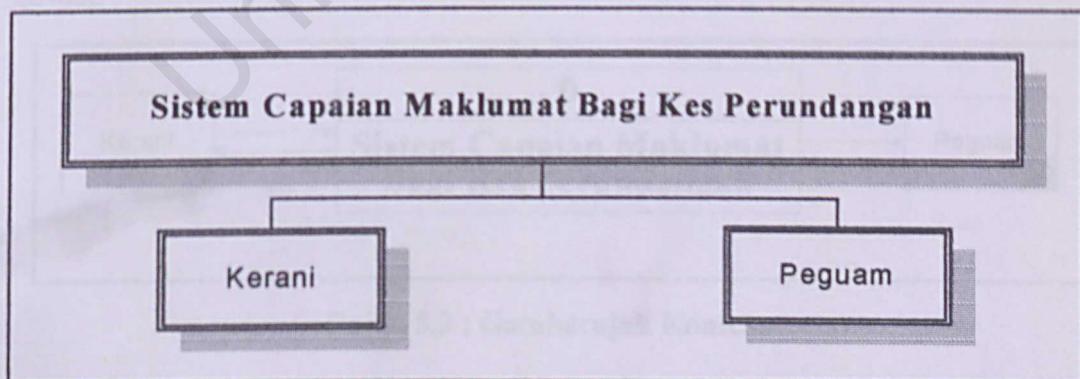
REKABENTUK SISTEM

BAB 5 : REKABENTUK SISTEM

Tujuan utama rekabentuk sistem ini dilakukan adalah untuk memastikan bahawa sistem yang bakal dibina adalah bertepatan dengan proses asal dan perancangan ini dapat menyelesaikan masalah yang bakal timbul semasa pembangunan sistem walaupun sistem itu belum dibangunkan (*Ian Sommerville 1995*). Justeru itu ia dapat meminimumkan ralat dan sebagai rujukan perbincangan tentang sistem yang bakal dibina. Dalam bab ini setiap sudut rekabentuk sistem pendaftaran kursus atas talian ini akan dibincangkan secara berperingkat.

5.1 MODEL SISTEM

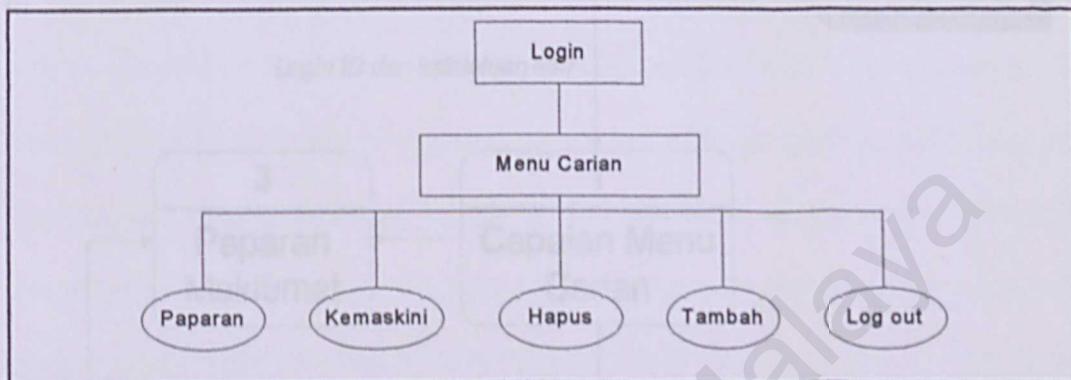
Struktur Sistem Capaian Maklumat Bagi Kes Perundangan dibahagikan kepada 2 mengikut tahap pengguna iaitu kerani firma dan peguam. Ini dapat dilihat pada Rajah 5.1 di bawah.



Rajah 5.1 : Struktur Sistem Capaian Maklumat Bagi Kes Perundangan

5.2 CARTA STRUKTUR

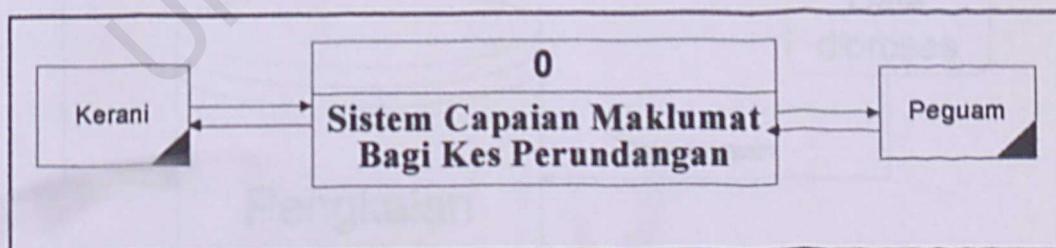
Carta struktur yang menunjukkan aliran fungsi yang boleh di akses oleh kerani dan peguam adalah sama kerana kedua-dua pengguna dibenarkan untuk mengakses kesemua halaman sistem tanpa halangan.



Rajah 5.2 : Carta Struktur Sistem

5.3 GAMBARAJAH KONTEKS

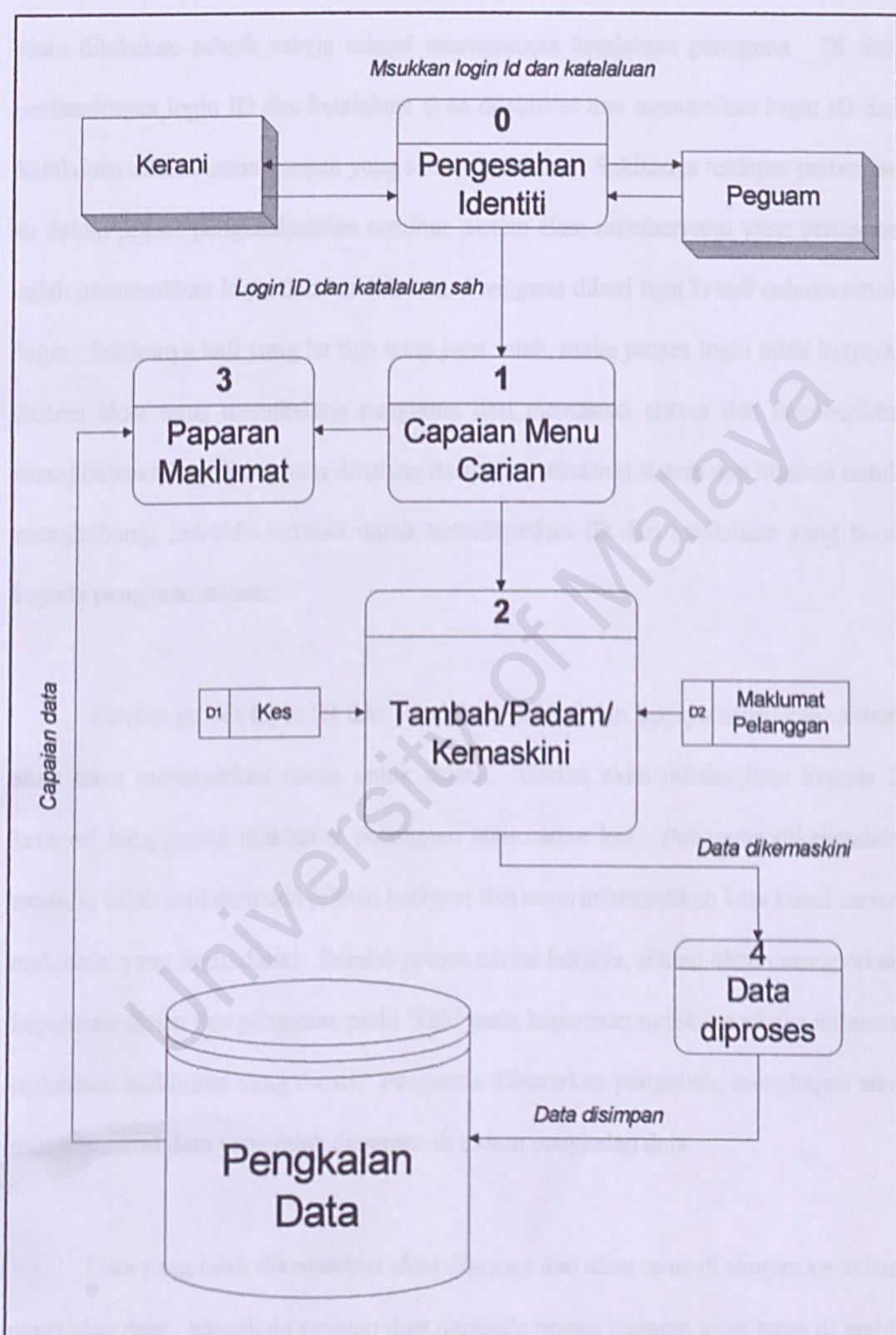
Secara keseluruhannya, Sistem Capaian Maklumat Bagi Kes Perundungan boleh dimodelkan dengan menggunakan gambarajah konteks seperti Rajah 5.3 di bawah.



Rajah 5.3 : Gambarajah Konteks

Gambarajah menunjukkan hubungan di antara pengguna sistem dan sistem. Ia menjadi asas dalam merekabentuk gambarajah data bagi sistem.

5.4 GAMBARAJAH ALIRAN DATA



Rajah 5.4 : Gambarajah aliran data bagi Sistem Capaian Maklumat Bagi Kes Perundangan

Berdasarkan gambarajah, pengguna iaitu kerani dan peguam dikehendaki memasukkan login ID dan katalaluan sebelum memasuki sistem. Pengesahan identiti akan dilakukan sebaik sahaja selesai memasukkan katalaluan pengguna. Di sini, perbandingan login ID dan katalaluan akan dilakukan dan memastikan login ID dan katalaluan adalah sama dengan yang telah ditentukan. Sekiranya terdapat perbezaan di dalam proses pengenalpastian tersebut, sistem akan memberitahu yang pengguna salah memasukkan login dan katalalauan. Pengguna diberi tiga(3) kali cubaan untuk login. Sekiranya kali yang ke tiga tetap juga salah, maka proses login tidak berjaya. Sistem akan terus menghalang pengguna dari memasuki sistem dan memberikan mesej bahawasanya pengguna dihalang daripada memasuki sistem dan diminta untuk menghubungi individu tertentu untuk memdapatkan ID dan katalaluan yang betul kepada pengguna sistem.

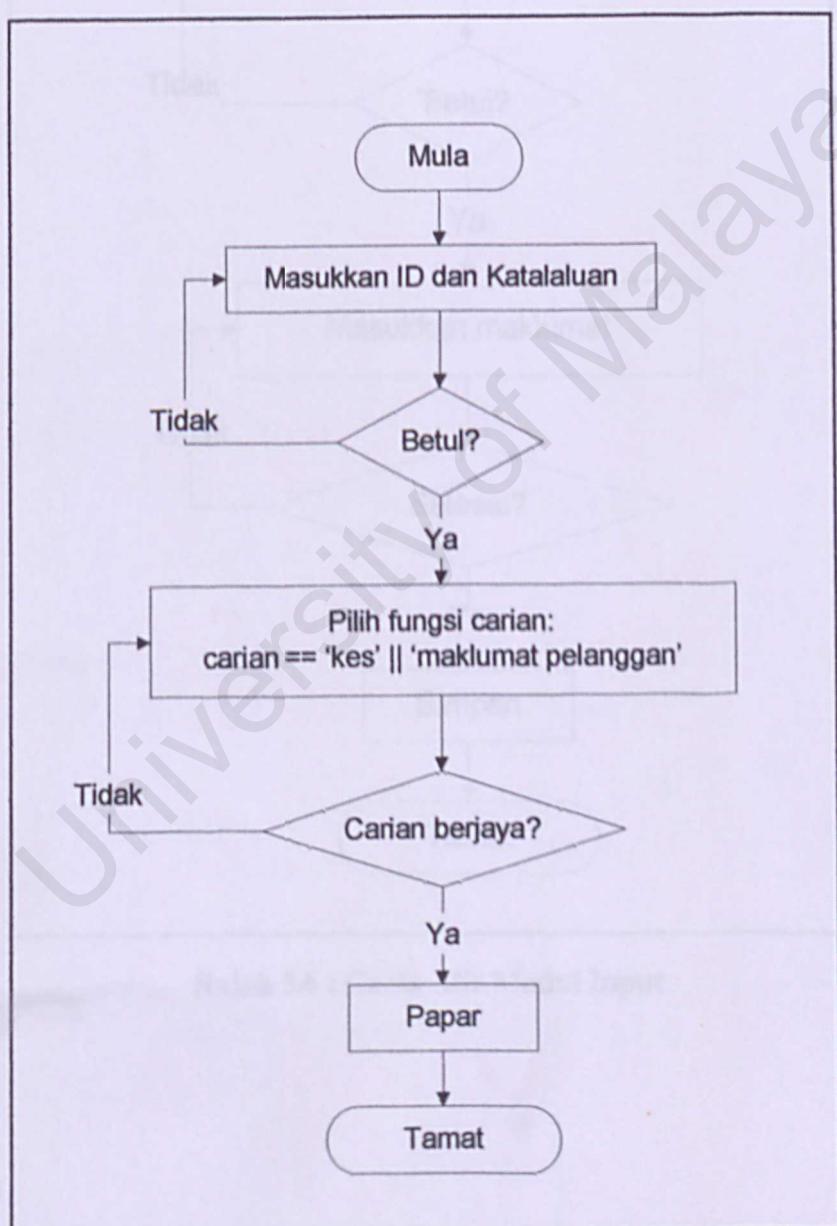
Setelah proses login ID dan katalaluan selesai dan berjaya dilakukan, sistem akan terus memaparkan menu untuk carian. Carian akan dibahagikan kepada 2 kategori iaitu carian maklumat pelanggan atau carian kes. Pengguna dikehendaki memilih salah satu daripada pilihan kategori dan terus memasukkan kata kunci carian maklumat yang ingin dicari. Setelah proses carian berjaya, sistem akan memaparkan keputusan carian dan pengguna perlu ‘klik’ pada keputusan untuk membuka halaman berkenaan maklumat yang dicari. Pengguna dibenarkan pengubah, menghapus atau mengemaskini data yang telah disimpan di dalam pengkalan data.

Data yang telah dikemaskini akan diproses dan akan terus di simpan ke dalam pengkalan data. Manakala capaian data daripada proses paparan akan terus di ambil daripada pengkalan data.

5.5 CARTA ALIR SETIAP MODUL SISTEM

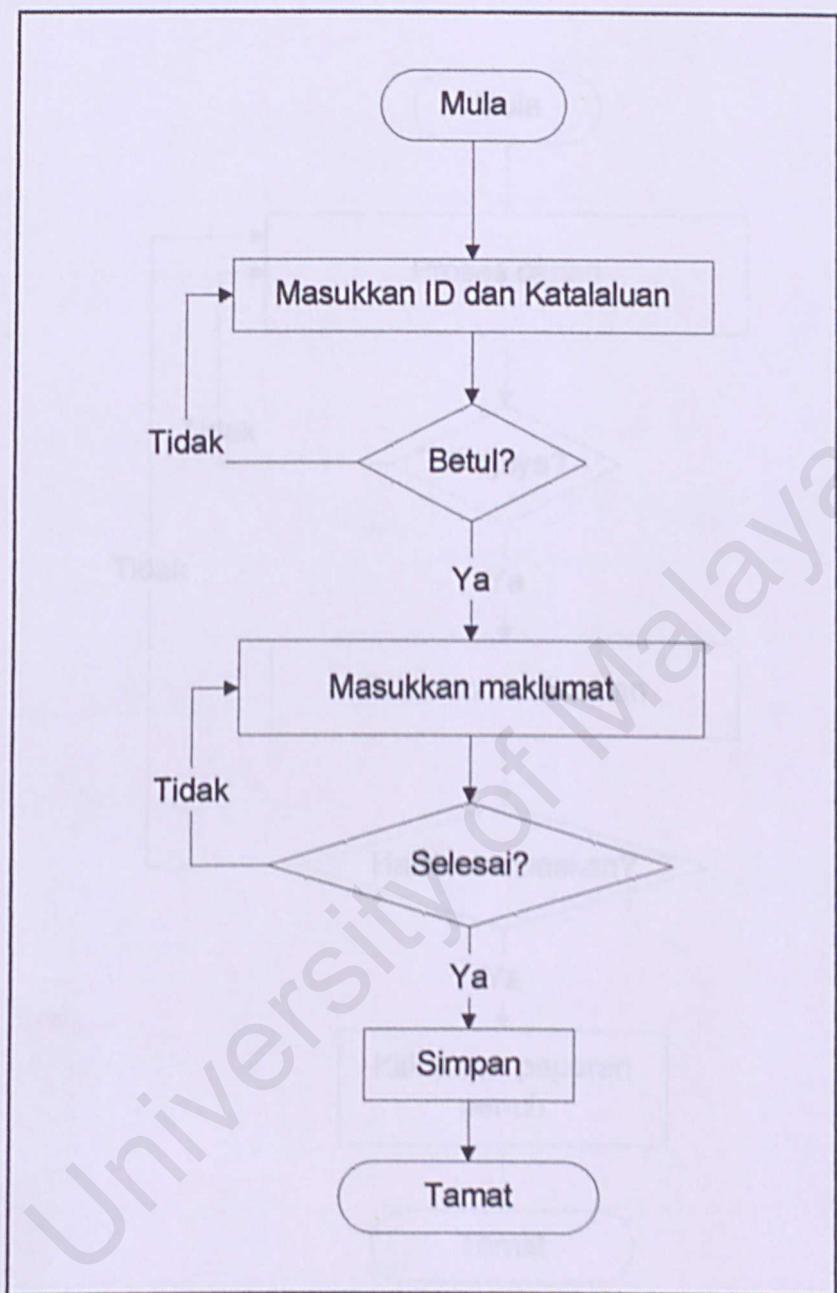
Setiap modul di dalam sistem ini mempunyai fungsi dan carta alir yang berlainan. Sistem mempunyai 3 modul iaitu carian, paparan dan input.

5.5.1 Carta Alir Modul Carian



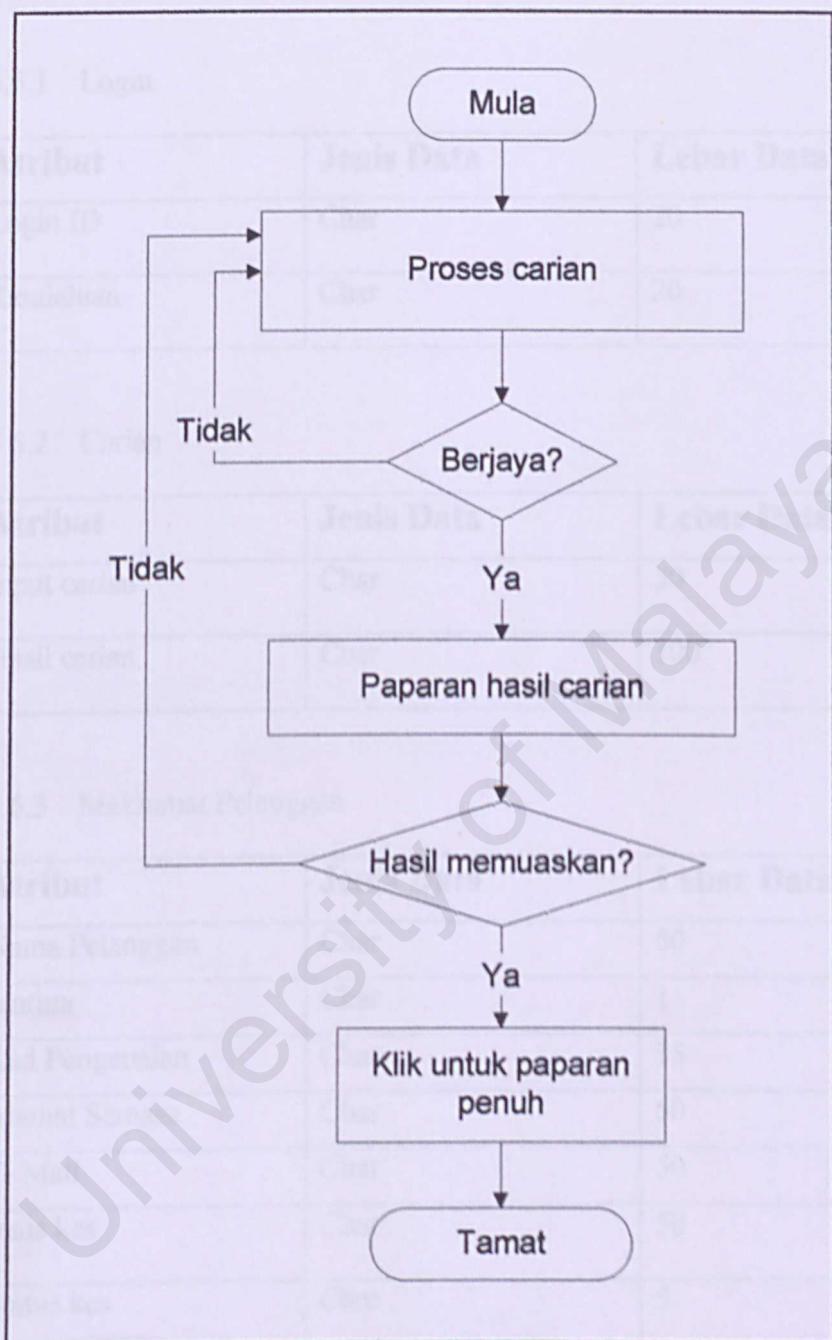
Rajah 5.5 : Carta Alir Modul Carian

5.5.2 Carta Alir Modul Input



Rajah 5.6 : Carta Alir Modul Input

5.5.3 Carta Alir Modul Paparan



Rajah 5.7 : Carta Alir Modul Paparan

5.6 KAMUS DATA SISTEM

5.6.1 Login

Atribut	Jenis Data	Lebar Data
Login ID	Char	20
Katalaluan	Char	20

5.6.2 Carian

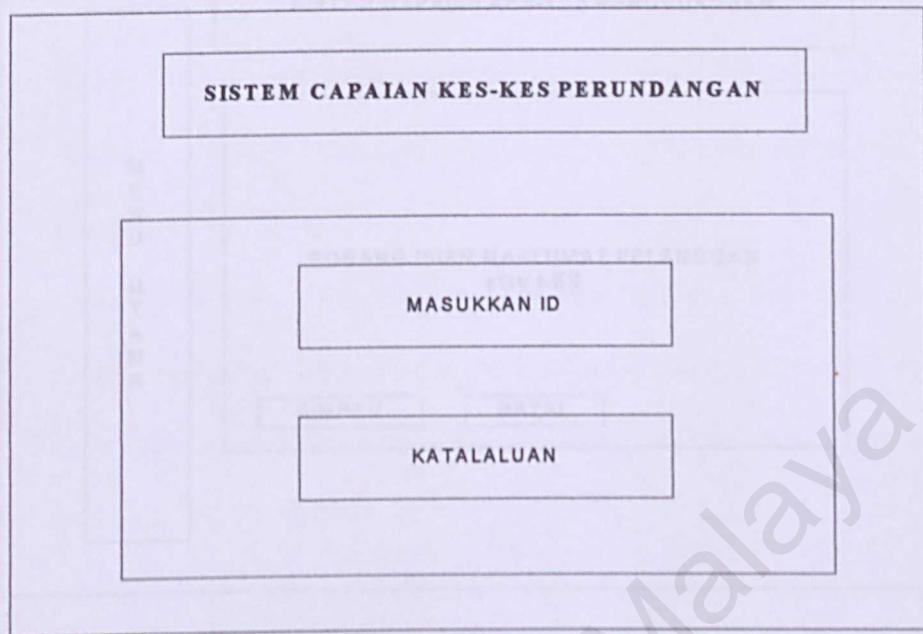
Atribut	Jenis Data	Lebar Data
Input carian	Char	30
Hasil carian	Char	100

5.6.3 Maklumat Pelanggan

Atribut	Jenis Data	Lebar Data
Nama Pelanggan	Char	50
Jantina	Char	1
Kad Pengenalan	Char	15
Alamat Semasa	Char	50
E-Mail	Char	50
Jenis kes	Char	50
Status kes	Char	5
Kes	Char	100

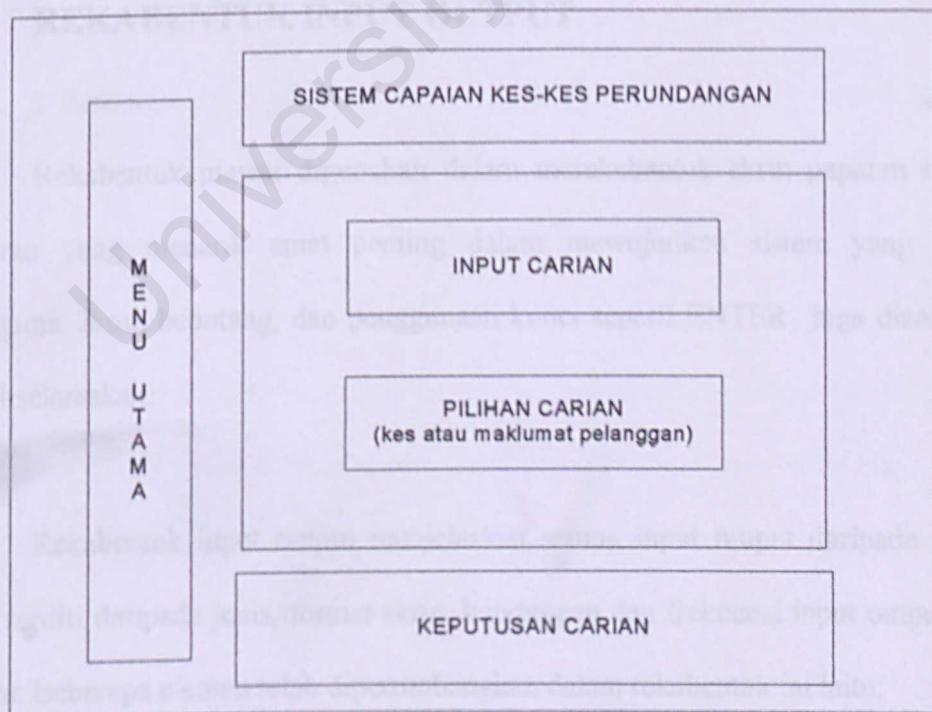
5.7 CADANGAN ANTARAMUKA SISTEM

5.7.1 Antaramuka Halaman Utama



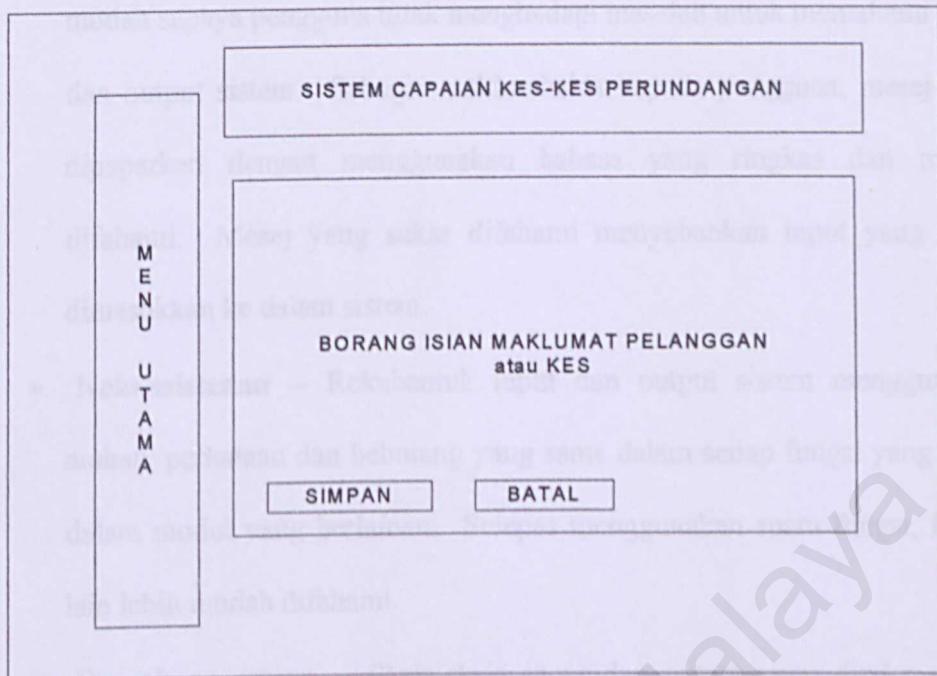
Rajah 5.8 : Halaman Utama

5.7.2 Antaramuka Modul Carian



Rajah 5.9 : Halaman Carian Data

5.7.3 Antaramuka Input



Rajah 5.10 : Halaman Input

5.8 REKABENTUK INPUT OUTPUT

Rekabentuk piawai digunakan dalam merekabentuk skrin paparan sistem.

Paparan yang menarik amat penting dalam mewujudkan sistem yang mesra pengguna. Ikon, bebutang, dan penggunaan kunci seperti ENTER juga disediakan dan diselaraskan.

Rekabentuk input output menjelaskan semua input output daripada sistem

yang terdiri daripada jenis, format skrin, kandungan dan frekuensi input output oleh sistem. Beberapa elemen telah dipertimbangkan dalam rekabentuk ini iaitu;

- **Kejelasan** -- Sistem akan menggunakan Bahasa Inggeris dan Melayu yang mudah supaya pengguna tidak menghadapi masalah untuk memahami input dan output sistem. Sebagai maklumbalas kepada pengguna, mesej turut dipaparkan dengan menggunakan bahasa yang ringkas dan mudah difahami. Mesej yang sukar difahami menyebabkan input yang salah dimasukkan ke dalam sistem.
- **Kekonsistenan** -- Rekabentuk input dan output sistem menggunakan arahan, perkataan dan bebutang yang sama dalam setiap fungsi yang sama dalam modul yang berlainan. Selepas menggunakan suatu fungsi, fungsi lain lebih mudah difahami.
- **Ramah pengguna** -- Skrin-skrin input dan output yang direka adalah berkonsepkan ramah pengguna. Contohnya sistem ini menyediakan paparan mesej ralat dan bantuan secara terus ketika pengguna menggunakan sistem. Penggunaan menu memudahkan pengguna memilih fungsi sistem yang dikehendaki. Walau bagaimanapun terdapat juga mod-mod yang menggunakan bebutang bagi capaian kepada mod-mod yang lain. Skrin pertanyaan dan pengesahan juga dibuat untuk sebarang perubahan atau penghapusan data yang dibuat.

Dalam tempoh pembangunan sistem ini, penilaian dan pengujian dalam fasa rekabentuk perlu dilakukan dengan lebih kerap dan memakan masa yang lama kerana banyak masalah yang timbul. Perubahan-perubahan dilakukan secara berperingkat untuk mendapatkan bentuk terbaik dan memastikan sistem dapat dilarikan dengan sempurna.

BAB 6

PELAKSANAAN SISTEM

BAB 6 : PELAKSANAAN SISTEM

6.1 PENGENALAN

Setelah rekabentuk input dan output secara manual dilakukan dan diikuti dengan rekabentuk piawai, pengkodan **Sistem Capaian Maklumat Bagi Kes Perundangan** dilakukan secara berperingkat iaitu dengan menyiapkan satu modul dan diikuti dengan modul lain. Pada akhir fasa, modul-modul dihubungkan antara satu dengan lain. Beberapa penekanan dibuat semasa menjalankan fasa ini iaitu :

6.1.1 Penyelenggaraan aturcara yang mudah.

Kod program yang dibangunkan dengan menggunakan perisian Active Server Pages (ASP) ini dibuat dengan menggunakan teknik pengkodan yang mudah difahami dengan penggunaan pembolehubah yang menyerupai nama medan. Komen untuk aturcara pula dibuat untuk memudahkan program ini difahami.

6.1.2 Faktor ketahanan

Faktor ketahanan diuji semasa fasa pengkodan dengan menggunakan data contoh untuk melihat kebolehan prosedur aturcara yang dibangunkan dengan menerima situasi yang tidak normal seperti input yang mempunyai jenis yang tidak sama atau di luar julatnya.

6.1.3 Faktor ramah pengguna.

Skrin output dan input piawai dibuat mengikut spesifikasi yang dibuat oleh pengguna. Mesej yang dipaparkan di sepanjang larian adalah berbentuk skrin mesej yang keluar sekiranya membuat input yang salah supaya pengguna tidak menghadapi masalah semasa menggunakan sistem.

Penggunaan butang, kekunci dan menu akan memudahkan lagi pengguna tanpa perlu mengingati proses yang telah dilakukan atau terpaksa menghafal langkah-langkah tertentu.

6.1.4 Faktor kecekapan.

Kod program yang ditulis adalah cekap dari segi penggunaan pembolehubah, prosedur dan fungsi yang dibuat. Kelebihan kod dalam program aturcara dielakkan supaya program dapat dilarikan dengan lebih cepat.

6.2 IMPLEMENTASI

Fasa implementasi dalam pembangunan sistem merupakan fasa di mana spesifikasi rekabentuk diterjemahkan kepada bahasa pengaturcaraan. Ini merupakan tanggungjawab pembangun untuk mengembangkan perincian sistem dari rekabentuk lakaran kepada aplikasi yang boleh dijalankan. Tujuan utama fasa implementasi ini adalah untuk menghasilkan kod sumber yang jelas, mudah berserta dokumentasi yang memudahkan proses pengujian, pengubahsuaian dan penyelenggaraan.

Selain itu, rekebentuk skrin juga direka di mana program antaramuka pengguna direka untuk berinteraksi dengan pengguna. Dalam mengimplementasikan sistem ini, rekabentuk antaramuka pengguna telah direka terlebih dahulu pada skrin berbanding dengan kod program. Ini adalah untuk memudahkan pembangun mendapat gambaran yang lebih jelas lagi terhadap sistem yang dibangunkan disamping mudah bagi pembangun menulis kod program. Terdapat banyak perubahan terhadap antaramuka pengguna semasa proses pelaksanaan di mana keperluan sentiasa berubah.

6.2.1 Perubahan Dalam Perisian Yang Digunakan

Pelaksanaan sistem ini melibatkan beberapa jenis perisian yang telah dijelaskan di dalam bab 2. Akan tetapi, terdapat penambahan perisian yang digunakan di dalam mengimplementasikan sistem ini iaitu ***Macromedia Dreamweaver UltraDev 4.0***.

Melalui *Dreamweaver UltraDev 4.0*, rekabentuk antaramuka sistem dan kod program dilakukan di sini. Rekabentuk adalah mudah dilaksanakan dengan beberapa fungsi yang telah disediakan di dalam sistem tanpa perlu menulis kod program untuk fungsi-fungsi tersebut. Fungsi '*click and drag*' adalah di antara fungsi yang dibekalkan di dalam perisian dan ini sangat memudahkan tugas pembangun sistem. Pengguna juga boleh melihat hasil antaramuka yang direka tanpa perlu melarikannya di platform lain. Perisian ini juga boleh menyambungkan sistem kepada pengkalan data. Pembangun tetap memilih Microsoft Access 2000 setelah mengambil kira saiz dan ruang data yang akan ditempatkan di pengkalan data. Pembangun boleh memasukkan data-data ke dalam pengkalan data tersebut dengan mudah. Fail-fail *.asp juga boleh ditulis dan dilarikan melaluinya dengan mudah.

6.2.2 Perubahan Dalam Skop Sistem

Di dalam skop yang telah disebutkan di dalam bab-bab yang terdahulu, proses memasukkan rekod pengguna adalah meliputi data-data seperti jenis kes, data peribadi anak guam, akta-akta yang berkaitan dengan dan juga apa-apa pindaan terkini kepada akta-akta tersebut (*Latest Amendment Of The Acts*). Namun begitu, setelah proses perlaksaan

dilakukan, terdapat perubahan di mana, data mengenai akta-akta yang berkaitan dengan dan juga apa-apa pindaan terkini kepada akta-akta tersebut (*Latest Amendment Of The Acts*) tidak akan dilaksanakan. Ini adalah permintaan daripada pihak pengguna/penguji sistem sendiri yang terdiri daripada kalangan peguam dan kerani firma guaman. Akta adalah satu bahan cetakan atau rujukan yang sentiasa diperlukan di dalam setiap kes dan pihak guaman menggunakanannya di dalam setiap proses di mahkamah. Akta dikehendaki dibeli oleh pihak firma guaman sebelum boleh menggunakan akta tersebut dalam mana sessi perundangan. Akta adalah di dalam bentuk bahan bercetak dan mempunyai nombor akta yang sentiasa dirujuk. Masalah akan timbul sekiranya akta dilakukan di dalam sistem kerana tiga situasi:

1. Terlalu banyak akta dan pindaan akta yang akan dimasukkan. Memerlukan masa yang panjang untuk proses memasukkan data tersebut secara '*key in*'.
2. Walaupun pembangun sistem sedar akan fungsi peranti '*scanner*' di dalam memasukkan data tersebut, pihak pengguna/penguji sistem menolak memasukkan data dengan cara ini. Alasannya ialah, akta sentiasa digunakan dalam setiap sidang mahkamah di dalam bentuk cetakan daripada akta yang asal. Kaedah cetakan keluar(*print out*) daripada sistem tidak dapat diterima bagi mengelakkan keaslian data.
3. Pembangun sedar masalah keaslian data dapat diatasi dengan pembangunan sistem yang baik dan boleh dipercayai. Namun, pihak pengguna sekali lagi menolak dengan alasan, bahan cetakan

daripada sistem tidak boleh menjadi dokumen rujukan semasa perbicaraan sedang berlangsung. Di mana ia memerlukan salinan daripada buku akta yang asal untuk di masukkan ke dalam satu fail dokumen yang menjadi rujukan dan bahan yang perlu disimpan oleh mahkamah semasa perbicaraan sedang berlangsung.

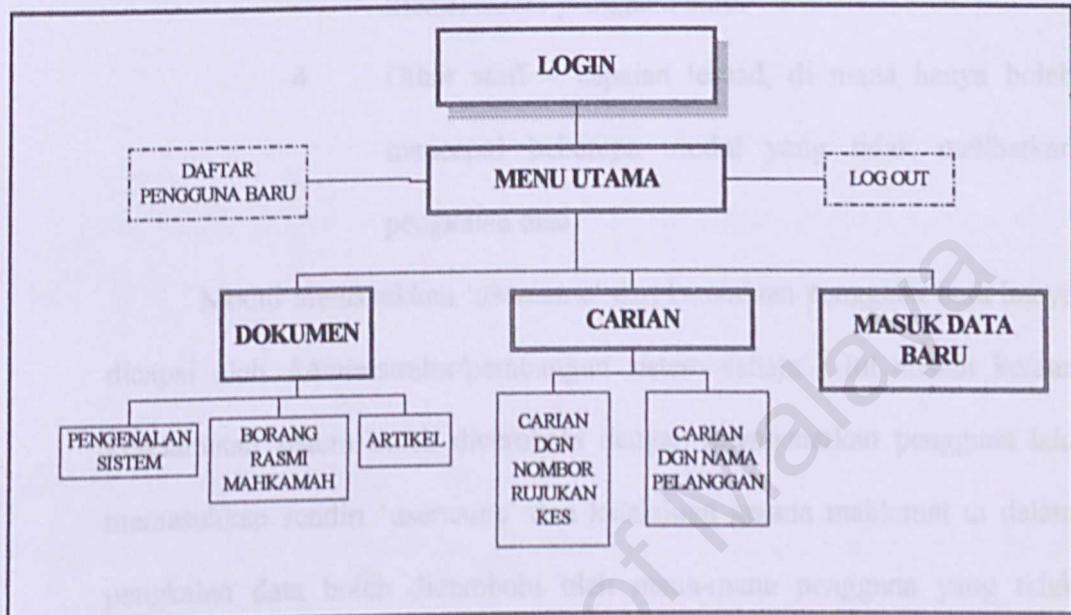
Setelah melihat kepada maklum balas daripada pihak pengguna sistem, maka pihak pembangun mengambil keputusan untuk membatalkan fungsi ini di lakukan di dalam sistem.

6.2.3 Perubahan Di Dalam Paparan Antaramuka

Perubahan paparan antaramuka adalah adalah disebabkan keperluan sistem yang menuntut satu antaramuka yang mudah dan dapat membantu pengguna untuk menggunakan sistem dengan mudah walaupun pada pertama kali. Pada awalnya, selepas antaramuka '*Login*' berjaya di lepasi, sistem akan terus memaparkan antarmuka '*Carian*'. Namun setelah melihat kepada keperluan sistem dan pengguna, paparan secara demikian adalah tidak relevan dan agak mengelirukan pengguna. Dengan itu, sebagai perubahan, selepas antaramuka '*Login*', sistem akan terus kepada antaramuka utama dan paparan antaramuka '*Carian*' hanya akan dipapar jika diminta oleh pengguna seperti antaramuka yang lain. Di samping itu, terdapat beberapa antaramuka paparan dan fungsian tambahan yang terdapat di dalam sistem yang tidak disebutkan di dalam bab Rekabentuk Sistem sebelum ini.

6.2.4 Perubahan Pada Carta Struktur

Perubahan carta struktur ini adalah berkait rapat dengan perubahan di dalam paparan antramuka sistem. Carta struktur yang telah diubah dapat dilihat pada Rajah 6.1 dibawah.



Rajah 6.1 : Carta Struktur Sistem Yang Telah Diubah

6.2.5 Perubahan di dalam faktor keselamatan.

Di dalam laporan fasa rekabentuk sistem sebelum ini, dinyatakan bahawa modul keselamatan sistem akan dilaksanakan dengan menghalang mana-mana pengguna yang salah memasukkan ‘username’ dan katalaluan sebanyak 3 kali cubaan. Namun, melihat kepada keperluan sistem, modul keselamatan tersebut telah diubah. Terdapat 4 peringkat pengguna di dalam sistem ini;

1. Administrator/pembangun sistem – capaian sistem secara penuh tanpa sebarang sekatan
2. Lawyer – capaian sistem penuh, namun masih tidak dibenarkan memasuki modul memasukkan pengguna

baru, yang hanya boleh dicapai oleh Administrator/pembangun sistem sahaja.

3. Clerk – capaian separa penuh, tidak oleh mencapai modul penambahan rekod, penghapusan rekod dan memasukkan pengguna baru.
4. Other staff – capaian terhad, di mana hanya boleh mencapai beberapa modul yang tidak melibatkan pengkalan data.

Modul memasukkan ‘username’ dan katalaluan pengguna baru hanya dicapai oleh Administrator/pembangun sistem sahaja. Ini adalah kerana keselamatan sistem boleh dicerobohi dengan membenarkan pengguna lain memasukkan sendiri ‘username’ dan katalaluan kerana maklumat di dalam pengkalan data boleh dicerobohi oleh mana-mana pengguna yang tidak dibenarkan. Oleh itu, sekiranya terdapat pengguna baru, hanya Administrator/pembangun sistem sahaja yang akan menguruskan proses kemasukan tersebut.

6.2.6 Perubahan di dalam skop carian

Carian maklumat pelanggan adalah berdasarkan 2 katakunci iaitu nombor rujukan kes dan nama pelanggan. Pengguna dikehendaki memilih sahaja daripada ‘list down menu’ yang akan memaparkan semua nombor fail rujukan dan nama pelanggan yang terdapat di dalam pengkalan data. Modul penambahan dan penghapusan rekod hanya akan dapat dilakukan selepas modul carian dilaksanakan.

6.3 PEMBANGUNAN

Berdasarkan kepada rekabentuk sistem yang dirancang dalam langkah terdahulu, sistem telah dibangunkan mengikut spesifikasi-spesifikasi tersebut. Pembangunan ini merangkumi beberapa segi:

6.3.1 Aturcara

Pembangunan Sistem Capaian Maklumat Bagi Kes Perundangan ini banyak melibatkan bahasa pengaturcara. Di antara bahasa yang digunakan ialah ASP, HTML, dan JavaScript. Ketiga-ketiga bahasa ini amat menyokong perlaksanaan sistem. Justeru itulah bahasa-bahasa ini dipilih .

6.3.2 Antaramuka

Kebanyakan rekabentuk antaramuka ini banyak mempertimbangkan ciri-ciri kekemasan dan keselesaan pengguna. Justeru itu hampir semua arahan pengguna dijana daripada menu atas dan hanya sedikit sahaja yang dijana daripada butang-butang.

6.3.3 Kapasiti Simpanan Data

Oleh kerana sistem ini bakal menguruskan ramai maklumat pelanggan maka kapasitinya dimaksimumkan. Namun jika berlaku sebarang permintaan daripada pihak pengurusan kapasiti tertentu boleh ditetapkan contohnya untuk menjimatkan ruang cakera keras.

6.3.4 Laporan

Tujuan sebenar sesebuah pangkalan data dibina adalah untuk memudahkan pemantauan dan pengurusan terhadap aktiviti kerja harian. Ia melibatkan banyak data mentah. Oleh itu ia perlu di analisis. Walaupun

sistem ini mampu mengeluarkan laporannya sendiri tetapi adalah lebih profesional jika rekabentuk laporan itu dapat dimanipulasi secara maksimum.

6.4 FUNGSI-FUNGSI UTAMA SISTEM CAPAIAN MAKLUMAT BAGI KES PERUNDANGAN

Terdapat beberapa fungsi-fungsi utama yang melaksanakan proses-proses di dalam sistem ini, antaranya ialah :

6.4.1 Pengesahan data (*Data validation*)

Sistem akan memberi paparan berjaya atau tidak berjaya setelah data dimasukkan, dikemaskini atau dihapuskan dari pengkalan data. Akan muncul skrin berlakunya pertindihan maklumat di dalam pengkalan data yang akan dilakukan di dalam fungsi memasukkan maklumat pengguna baru. Ini untuk memastikan “Username” pengguna tidak bertindih supaya tidak berlaku kekeliruan di dalam pengurusan pengkalan data.

6.4.2 Carian

Setiap capaian rekod yang dikehendaki oleh pengguna, fungsi carian bertindak untuk memaparkan rekod yang tersimpan di dalam pangkalan data.

6.4.3 Pengemaskinian

Fungsi ini hanya akan dilakukan setelah modul carian dijanakan. Rekod-rekod akan terus terpapar setelah berjaya dicapai dari pengkalan data. Pengguna dibenarkan menambah atau mengemaskini maklumat yang telah di masukkan.

6.4.4 Memasukkan Rekod

Fungsi memasukkan data baru ke dalam pengkalan data akan dilakukan dan akan muncul skrin pengesahan data telah dimasukkan ke dalam pengkalan data. Fungsi akan membenarkan pengguna memasukkan data berkaitan kes yang sedang dikendalikan oleh pengguna. Proses adalah di lalui dengan pengisian antaramuka ‘form’ yang telah disediakan tanpa apa-apa fungsi memasukkan data dengan cara yang khas.

6.4.5 Paparan Semula Rekod

Paparan dilakukan setiap kali pengguna selesai proses carian.

6.4.6 Penghapusan data

Juga dilakukan setelah proses carian dilakukan.

BAB 7

PENGUJIAN SISTEM

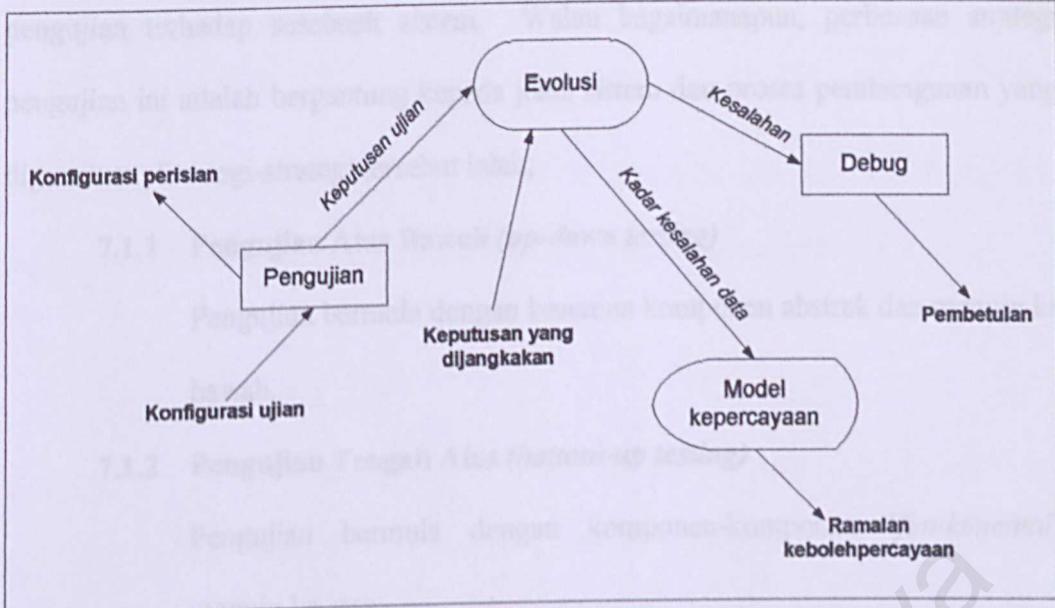
BAB 7 : PENGUJIAN SISTEM

Pengujian sistem merupakan satu unsur yang selalunya merujuk kepada pengesahan dan kesahihan (*verification and validation*). Pengesahan merujuk kepada suatu set aktiviti yang memastikan bahawa perisian melaksanakan suatu fungsi dengan betul dan menepati spesifikasi yang telah ditetapkan dan memenuhi keperluan dan kehendak pengguna. Kesahihan pula merujuk kepada set aktiviti yang berbeza yang memastikan sistem yang dibina memenuhi keperluan pengguna. Tujuan utama aktiviti pengesahan adalah untuk mencapai dan memperbaiki kualiti produk yang dihasilkan semasa pembangunan sistem [P. Sellapan, 2000].

Beberapa peraturan diaplikasikan bagi memenuhi objektif pengujian. Peraturan tersebut adalah [Pressman, 2001]:

- Pengujian merupakan suatu proses perlaksanaan program dengan tujuan untuk mendapat ralat.
- Pengujian yang berjaya ialah pengujian yang dapat mengenalpasti atau mendedahkan ralat yang tidak dapat didedahkan kemudiannya.
- Kes pengujian yang baik adalah kes yang mempunyai kebarangkalian yang tinggi mendapat ralat.

Jika pengujian dapat dilakukan dengan jayanya, ianya akan dapat mendedahkan ralat di dalam perisian di samping pengujian dapat menunjukkan sesuatu fungsi perisian yang dilaksanakan berdasarkan kepada spesifikasi. Rajah 7.1 menunjukkan aliran maklumat pengujian.



Rajah 7.1: Aliran Maklumat Pengujian

Semua aplikasi program yang baru ditulis atau diubahsuai mestilah diuji dengan betul. Pengujian “*trial and error*” adalah tidak memadai. Pengujian sebenar merupakan suatu proses yang berterusan. Ia dibuat sepanjang pembangunan sistem. Ia seharusnya mengenalpasti masalah bukannya menunjukkan kebaikan program. Walaupun pengujian merupakan proses membosankan tapi ia merupakan langkah yang penting bagi memastikan kualiti sebenar sistem yang dibuat.

7.1 PERINGKAT PENGUJIAN

Peringkat pengujian melibatkan penyediaan data-data untuk mengawal kesilapan setiap modul aturcara serat kegiatan mengawal atau menesan ralat logik di dalam setiap modul aturcara. Peringkat ini dilaksanakan dengan tujuan mengesahkan bahawa kesemua komponen sistem tidak mengandungi ralat. Terdapat 5 strategi

pengujian terhadap sesebuah sistem. Walau bagaimanapun, perbezaan strategi pengujian ini adalah bergantung kepada jenis sistem dan proses pembangunan yang digunakan. Strategi-strategi tersebut ialah;

7.1.1 Pengujian Atas Bawah (*up-down testing*)

Pengujian bermula dengan kesemua komponen abstrak dan menuju ke bawah.

7.1.2 Pengujian Tengah Atas (*bottom-up testing*)

Pengujian bermula dengan komponen-komponen ‘fundamental’ menuju ke atas.

7.1.3 Pengujian “Thread” (*thread testing*)

Digunakan untuk sistem pelbagai pemprosesan di mana proses pemindahan “thread” melalui proses-proses ini.

7.1.4 Pengujian Tekanan (*stress testing*)

Di mana kepercayaan penekanan terhadap sistem dengan melalui had yang telah ditetapkan dan pengujian sistem dengan melalui had yang telah ditetapkan dan pengujian bagaimana sistem boleh dicapai dalam pelbagai situasi.

7.1.5 Pengujian belakang ke belakang (*back-to-back testing*)

Yang digunakan apabila versi sistem telah sedia ada. Sistem diuji bersama dan outputnya dibandingkan.

7.2 PROSES PENGUJIAN

Proses pengujian yang digunakan di dalam sistem ini terdiri daripada pengujian unit, pengujian modul, pengujian integrasi dan pengujian sistem.

7.2.1 Pengujian Unit

Di dalam pengujian unit, setiap unit aturcara diuji bersendirian. Ujian dilakukan dengan menggunakan set-set data ujian yang ditentukan dan hasilnya diperhatikan. Ini membolehkan unit-unit berfungsi dengan jenis input yang dicadangkan.

7.2.2 Pengujian Modul

Oleh kerana proses pembangunan sistem ini menggunakan kaedah air terjun, maka modul-modul aturcara dilaksanakan dari peringkat atas kemudian diuji setiap kali satu fungsi baru ditambahkan kepadanya. Dengan ini, pembangun dapat mengesan ralat dengan mudah semasa larian sistem.

7.2.3 Pengujian Integrasi

Di dalam fasa ini, ujian dilakukan ke atas antaramuka 2 komponen yang berinteraksi di dalam sesuatu unit. Jadi, satu ralat yang timbul menyebabkan pelayan tidak dapat mengkompilasi fail tersebut dengan jaya. Jadi, proses ini harus dilakukan dengan teliti dengan memastikan bahawa sistem dapat berintegrasi dengan baik. Terdapat 2 jenis integrasi iaitu **Integrasi Atas Bawah** dan **Integrasi Bawah**.

Atas. Pada proses ini, pengujian ke atas penghantaran parameter juga dilakukan. Pengujian perhubungan dengan pengkalan data juga dilakukan bagi memastikan sistem dapat berinteraksi dengan pengkalan data dengan baik.

7.2.4 Pengujian Sistem

Pengujian sistem bermula setelah aturcara-aturcara berjaya dilarikan tanpa ralat di dalam pengujian intergasi. terbahagi Objektif-objektif ujian ini adalah untuk

7.2.4.1 Mengesahkan ketepatan dan kejituuan semua komponen sistem yang dibangunkan, berdasarkan spesifikasi-spesifikasi sistem yang telah direkabentuk. Setiap subsistem dipastikan akan boleh dilarikan dengan baik. Sistem ini sepatutnya beroperasi sebagaimana yang dikehendaki dalam keadaan yang sama dengan persekitaran operasi yang sebenar.

7.2.4.2 Mengukur prestasi sistem, pada keseluruhannya, sama ada ia dapat mencapai tahap yang boleh diterima.

7.2.4.3 Mengukur sejauh mana sistem yang dibangunkan itu dapat memenuhi objektif-objektif yang telah ditentukan. Untuk melaksanakan proses ini, sejumlah data telah dimasukkan ke dalam pengkalan data untuk menguji kebolehlarian sistem. Selain itu, data-data yang berbeza cuba

dimasukkan ke dalam borang supaya dapat menguji integriti sistem.

7.2.5 Pengujian Pencapaian

Pengujian pencapaian adalah untuk keperluan yang bukan fungsi terhadap sesuatu aplikasi. Jenis-jenis pencapaian ujian percapaian yang terlibat dalam sistem ini adalah :

7.2.5.1 Ujian data dan rekod

Ujian terhadap medan dan rekod diperiksa samada ia boleh menerima segala kemungkinan data dari pengguna.

7.2.5.2 Ujian keselamatan

Ujian ini adalah untuk memastikan bahawa aplikasi sistem yang dihasilkan memenuhi keperluan keselamatan. Beberapa ujian dijalankan untuk mengetahui samada sistem boleh dicerobohi oleh pengguna yang tidak sah. Sekiranya sistem boleh dicerobohi, kaedah keselamatan lain perlu dipertimbangkan.

7.2.5.3 Ujian masa

Pencapaian sistem diambil masa untuk memastikan ianya memenuhi keperluan pengguna. Ujian ini dilakukan semasa masa larian(*run-time*) untuk memastikan prestasi persembahan sistem secara keseluruhan. Ini termasuklah dari segi tindakbalas, ingatan yang digunakan dan kecekapan sistem.

7.2.5.4 Ujian faktor kemanusiaan

Antaramuka dan mesej diperiksa untuk memastikan bahawa aplikasi sistem mempunyai cirri-ciri mesra pengguna.

7.2.5.5 Ujian baikpulih

Ujian ini dijalankan bertujuan menggagalkan sistem dan memastikan kegagalan tersebut dapat dipulihkan semula samada ianya dilakukan secara automatik oleh sistem ataupun berdasarkan masukan input pengguna.

7.3 JENIS-JENIS RALAT DAN KESALAHAN

Sepanjang pembangunan dan pelaksanaan sistem, pengujian yang berterusan perlu dilakukan terhadap sistem bagi memastikan sistem yang dibangunkan adalah konsisten dan bebas dari ralat. Terdapat 3 jenis ralat dan 2 jenis kesalahan yang biasa ditemui;

7.3.1 Ralat Kompilasi

Ralat ini terjadi apabila kod yang dibina salah disebabkan kesilapan menaip kata kunci atau tertinggal tanda-tanda yang penting. Ralat-ralat sintak juga termasuk dalam jenis ralat ini.

7.3.2 Ralat Masa Larian

Ralat ini berlaku apabila perlaksanaan sistem cuba melakukan sesuatu operasi yang tidak boleh dilaksanakan oleh sistem.

7.3.3 Ralat Logik

Ralat logik berlaku apabila operasi yang diperuntukkan kepada aplikasi tidak menghasilkan keputusan seperti yang dikehendaki.

Keadaan ini berlaku walaupun kod yang sah telah diperuntukkan kepada perlaksanaan sistem.

7.3.4 Kesalahan Algoritma

Terjadi apabila komponen algoritma atau logik tidak menghasilkan output yang baik untuk input yang telah diberikan oleh kerana berlaku sesuatu kesilapan semasa langkah pemprosesan. Kesalahan ini mudah untuk dikenalpasti dengan melihat kepada aturcara (*fall desk checking*) atau dengan menghantar data input kepada setiap data kelas yang berlainan. Masalah ini kerap terjadi apabila menggunakan bahasa pengaturcaraan Active Server Pages (ASP) untuk menulis aturcara kerana kebanyakan pengaturcara lupa untuk melengkapkan aturcara mereka. Jenis-jenis kesalahan algoritma adalah :

- Ujian yang salah terhadap syarat pilihan.
- Terlupa untuk mengistiharkan pembolehubah.
- Gelung tidak konsisten.
- Terlupa untuk menguji terhadap kes-kes rekod tertentu (contoh pembahagian rekod dengan nilai kosong)

7.3.5 Kesalahan Sintak

Kesalahan atau kesilapan boleh diperiksa semasa berlakunya kesilapan algoritma. Ini akan menyebabkan penulisan sesuatu bahasa

pengaturcaraan tidak digunakan dengan tepat. Bagi arahan di dalam JavaScript, kesalahan sintak dapat dikesan dengan mudah kerana ia dilarikan berdasarkan baris demi baris. Justeru itu pengaturcara menggunakan editor yang dapat memberi koordinat semasa untuk mengesan sebarang kesalahan sintak. Selepas sesuatu program dijalankan, jika berlaku sebarang kesalahan sintak ia akan dapat dikenalpasti dan memaklumkan jenis kesilapan serta dimana lokasinya.

BAB 8

PERBINCANGAN

BAB 8 : PERBINCANGAN

8.1 PENILAIAN SISTEM

Penilaian sistem dilakukan selepas ianya diberikan kepada pengguna untuk melakukan operasi yang telah ditentukan. Suatu jangkamasa tertentu perlu diberikan kepada pengguna untuk mereka membiasakan diri dengan sistem dan untuk memberi peluang kepada sistem itu untuk berada dalam keadaan stabil.

Tujuan utama peringkat ini ialah supaya pembangun boleh menilai samada sistem yang dibangunkan itu memenuhi objektif yang telah ditetapkan. Ia membandingkan sistem baru itu dengan jangkaan kelancaran yang diharapkan dan memerhatikan perkara-perkara tidak dirancang yang berlaku. Perkara-perkara yang negatif akan dicatat dan diperbetulkan, manakala perkara positif (*dalam bentuk fungsi penggunaan*) diambil perhatian. Ini merupakan peringkat terakhir proses pembangunan sistem.

8.2 KELEBIHAN SISTEM

Sistem ini mempunyai beberapa kelebihan seperti dinyatakan di bawah.

8.2.1 Antaramuka Pengguna

Antaramuka yang ramah iaitu pengguna hanya perlu pengetahuan minima tentang penggunaan papan kekunci dan tetikus. Disamping itu, fungsi-fungsi diberi dalam bentuk ayat di mana pengguna hanya perlu membaca dan memberi respon yang sepatutnya. Oleh itu tiada latihan mendalam diperlukan untuk menggunakan sistem ini. Menu pilihan disusun secara konsisten untuk memudahkan pengguna beralih dari satu menu ke

menu yang lain. Penggunaan warna dibuat secara minima dalam sesuatu borang atau paparan pada skrin. Setiap halaman juga disertakan dengan warna latar belakang yang sama supaya tidak mengelirukan pengguna dan mengurangkan kejutan dari pengguna. Mesej-mesej ringkas dengan deskripsi yang sesuai dipaparkan apabila berlaku ralat misalnya pengguna memasukkan katalaluan yang tidak sah. Mesej yang sesuai juga akan dipaparkan kepada pengguna apabila berlaku kesalahan dalam pangkalan data.

8.2.2 Mudah Digunakan

Pengguna boleh menggunakan ini dengan mudah dan selesa selain daripada bar navagasi yang tersedia dalam perisian pelayar. Pengguna juga boleh membuat pilihan untuk ke laman-laman utama melalui menu yang telah disediakan di sebelah atas setiap halaman.

8.2.3 Integriti Data

Rekabentuk pangkalan data yang baik seperti Microsoft Access 2000 memastikan integriti berlaku antara data. Aturcara kawalan yang dibina juga membantu memastikan integriti data serta keselamatannya. Contoh integristi data ialah seperti aplikasi mengelakkan berlakunya pertindihan data iaitu hanya satu data unik yang dimasukkan dalam pangkalan data dan tidak berlaku penduaan data yang sama didalamnya serta memastikan hanya data yang sah sahaja disimpan dalam satu-satu medan.

8.2.4 Laman Pengesahan

Ini merujuk kepada laman yang dipaparkan sebaik sahaja pengguna menghantar borang atau data yang diinput bagi mengesahkan bahawa maklumat yang dihantar telah dimasukkan ke dalam pangkalan data.

8.2.5 Keselamatan

Untuk memastikan keselamatan maklumat, pengesahan pengguna digunakan untuk mengelakkan pengguna yang tidak sah mencerobohi sistem.

8.2.5.1 Keselamatan Halaman

Individu diberi katalaluan untuk tujuan mentadbir dan katalaluan ini boleh ditukar mengikut keperluan tertentu. Katalaluan yang dimasukkan disemak terlebih dahulu kesahihannya pada pangkalan data sebelum diberi akses. Disamping itu, hanya pentadbir tertentu sahaja yang dibenarkan untuk membuat penambahan, pengemaskinian dan penghapusan rekod pentadbir lain.

8.2.5.2 Kerahsiaan Kod

Perlaksanaan kod aturcara ASP akan dilakukan dibahagian pelayan, maka kod dipaparkan di bahagian pengguna adalah dalam format HTML, dengan itu kerahsiaan kod adalah selamat untuk sistem ini.

8.3 KEKANGAN DAN PENYELESAIAN

Dalam melaksanakan pembangunan sistem, pembangun tidak terlepas daripada melaksanakan beberapa masalah. Antara masalah-masalah yang dihadapi oleh pembangun semasa proses implementasi serta langkah-langkah penyelesaian yang diambil untuk menangani masalah-masalah tersebut diterangkan seperti di bawah.

8.3.1 Kurang Pengetahuan Dalam Bahasa Pengaturcaraan.

Masalah: Kurang penguasaan dalam pengaturcaraan ASP. Masalah ini menyebabkan proses pembangunan sistem ini agak perlahan daripada yang dirancangkan. Ini kerana banyak aspek yang perlu diambilkira dalam pembinaan sistem.

Penyelesaian: Menggunakan utiliti yang berbeza yang mudah diintegrasikan.

Oleh kerana saya mempunyai sedikit kemahiran dan pengalaman dalam membangunkan sistem ketika menjalani latihan industri, segala pengetahuan itu saya gunakan sepenuhnya. Saya juga banyak melakukan rujukan dari buku-buku berkaitan ASP dan juga banyak bertanya dan berbincang dengan rakan-rakan dan pensyarah yang bijak di dalam ASP.

8.3.2 Kesukaran Membuat Pilihan/Keceluaran

Masalah: Pembangun juga menghadapi masalah keceluaran di awal pengemplimentasian sistem ini, di mana pembangun tidak tahu sama ada hendak menggunakan perisian Microsoft FrontPage atau Macromedia Dreamweaver UltraDev dalam rekebentuk

antaramuka dan menulis kod program. Pada awalnya, pembangun tidak tahu menggunakan perisian Macromedia Dreamweaver UltraDev, tetapi pembangun melihat perisian tersebut mempunyai banyak kelebihan berbanding FronPage.

Penyelesaian: Pembangun akhirnya mengambil keputusan memilih Macromedia Dreamweaver UltraDev di dalam membangunkan sistem ini. Walaupun pembangun tidak mengetahui cara penggunaannya, tetapi pembangun telah merujuk kepada buku berkaitan Macromedia Dreamweaver UltraDev, perbincangan dan soal jawab dengan rakan-rakan.

8.3.3 Tuntutan Akademik

8.3.4 Tuntutan Akademik Dan Kesuntukan Masa

Masalah: Selain menyelesaikan tugasan latihan ilmiah, saya juga terpaksa mengambil berat akan tugas-tugas bagi subjek-subjek lain yang diambil pada semester ini. Ini bermakna pembangun terpaksa membahagikan masa di antara tugasan latihan ilmiah dengan tugas-tugas subjek lain memandangkan faktor masa yang singkat. Walaubagaimanapun, ianya mungkin sedikit sebanyak menjelaskan fasa pembangunan sistem. Mungkin hal yang sama berlaku di mana kelewatan pembangunan dalam memulakan proses pembangunan sistem ini.

Penyelesaian: Pengurusan masa sistematik penting dalam memastikan kedua-dua jenis tugasan tersebut dapat diselesaikan dengan baik. Memikirkan keadaan rutin harian yang ada, langkah

kesemasaan yang dilakukan adalah sering kali berhubung dengan rakan-rakan bagi mendapat pandangan atau idea-idea yang berhubungkait dengan pembangunan sistem. Nampaknya, ia bukan sahaja memudahkan pembangunan tetapi ia juga akan melicinkan perjalanan projek pembangunan. Sebagai inisiatif, pembangun juga berharap untuk dirinya agar dapat memulakan sesuatu kerja itu seawal mungkin dengan mempunyai perancangan yang strategik.

8.3.5 Ralat Di Luar Jangkaan

Masalah: Ralat di luar jangka yang dimaksudkan di sini adalah keadaan sistem pengendalian yang ada dalam sesuatu komputer tidak berjalan dengan baik. Senagai contoh, '*illegal operation*', '*invalid path*' dan sebagainya. Ini menyebabkan pembangun terpaksa membuat kemaskini terhadap fail-fail sistem komputer yang menyebabkan ralat tersebut. Amnya, penangguhan dalam membangunkan sistem juga turut berlaku.

Penyelesaian: Masalah ini dapat diatasi dengan menemaskini fail yang menghasilkan ralat tersebut. Selain itu, bantuan rakan-rakan juga diperlukan bagi memberi pandangan terhadap gangguan ralat yang berlaku.

8.4 CADANGAN PERLUASAN SISTEM

Secara keseluruhannya, sistem ini masih boleh diperluaskan skop dan objektifnya. Ia dapat diperkuatkan pula dengan tambahan-tambahan idea baru, namun, terdapat beberapa aspek yang boleh ditambah dan diperbaiki bagi menjadikan sistem ini lebih menarik, mantap dan lebih fleksibel. Di antara aspek-aspek atau fungsi yang dirasakan perlu ditambah bagi perubahan di masa hadapan ialah antaranya :

- Skop sistem juga boleh ditambah agar tahap keberkesanan dan kesempurnaan sistem ditingkatkan. Contohnya, sistem ini tidak hanya memfokuskan kepada Undang-Undang Kekeluargaan sahaja tetapi mencakupi keseluruhan akta atau undang-undang dalam sivil atau syariah.
- Sistem ini juga boleh dijadikan sebahagian daripada submodul kepada sistem yang telah sedia ada. Terdapat banyak sistem yang dihasilkan oleh pembangun sistem berkenaan perundangan, sekiranya ia dapat diintegrasikan, ia dapat menghasilkan kepelbagaian dan pembaharuan dalam sistem.

8.5 KESIMPULAN PERBINCANGAN

Sistem ini telah berjaya mencapai objektif asanya walaupun terdapat beberapa kekurangan yang tidak dapat dielakkan dan ianya juga memenuhi keseluruhan keperluan fungsian dan bukan fungsian. Pengalaman terlibat di dalam pembangunan sistem ini merupakan suatu yang amat berharga dan amat mencabar serta memerlukan ketekunan.

SARANAN

KESELURUHAN SISTEM

SARANAN KESELURUHAN SISTEM

Reputasi sesuatu sistem itu bergantung kepada perancangan dan pengurusan yang baik sama ada seseorang pembangun itu menilai sistem yang sedia ada atau menganalisa segala keperluan fungsian yang akan digunakan. Di dalam hal ini, pertimbangan diberikan ke atas saranan yang telah dikemukakan agar dapat memperolehi keberkesanan sistem itu lebih baik. Sebagai satu sistem yang pertama kali diwujudkan, pengenalpastian masalah yang timbul pada peringkat awal agak kurang dititikberatkan. Sebaliknya ia lebih menjurus kepada cabang penyelesaian kepada permasalahan yang mungkin wujud. Oleh yang demikian, tidak dinafikan bahawa masalah yang sama akan timbul dalam fasa pembangunan yang berikutnya dan dari situlah penjanaan set alternatif yang dapat digunakan bagi mengurangkan risiko yang wujud. Sememangnya projek akhir yang dilakukan oleh setiap pelajar Sarjana Muda Sains Komputer Dan Teknologi Maklumat ini memerlukan perhatian dan komitmen daripada pelajar sendiri serta sokongan dan dorongan daripada penasihat bagi memastikan projek disiapkan pada masa yang ditetapkan.

KESIMPULAN

KESELURUHAN SISTEM

University of Malaya

KESIMPULAN KESELURUHAN SISTEM

Sistem Capaian Maklumat Bagi Kes Perundangan ini telah berjaya dibangunkan berdasarkan spesifikasi-spesifikasi yang telah ditetapkan. Namun ia masih lagi di peringkat permulaan pengimplementasian. Oleh itu ia berkemungkinan untuk berlaku ralat kerana pengujian belum dilakukan secara menyeluruh dan ia hanya akan mencapai seratus peratus memenuhi kehendak pelanggan apabila ia telah digunakan kelak. Diharapkan Sistem Capaian Maklumat Bagi Kes Perundangan ini dapat membantu para peguam dalam menagani segala masalah berkaitan simpanan dan carian maklumat perundangan. Sebarang usaha untuk mengembangkannya dapat diteruskan supaya sistem ini sentiasa bersesuaian dengan kehendak pengguna.

LAMPIRAN

University of Malaya

KAJIAN SOAL SELIDIK SISTEM CAPAIAN KES PERUNDANGAN

Nama firma guaman: _____

Jawatan anda: _____

Maklumat dan identiti akan dirahsiakan.

Tandakan jawapan anda

1. Berapa buah komputer yang terdapat di firma anda?

- a. Tiada
- b. Kurang dari 5
- c. Lebih dari 5
- d. Lebih dari 10

2. Adakah anda biasa menggunakan komputer?

- a. Ya
- b. Tidak

Tandakan jawapan anda : 5 - paling banyak; 4 - banyak; 3 - sederhana; 2 - sedikit;

1 - tiada.

3. Di manakah anda menyimpan maklumat mengenai pelanggan-pelanggan anda?

- a. fail
- b. disket
- c. compact disc
- d. ingatan komputer

4. Di manakah anda menyimpan maklumat mengenai kes-kes?

- a. fail
- b. disket
- c. compact disc
- d. ingatan komputer

5. Di manakah segala maklumat mengenai pengguna dan kes-kes disimpan?

- a. Di dalam satu tempat/bilik khas
- b. Disimpan sendiri oleh setiap kakitangan
- c. Disimpan oleh kerani
- d. Disimpan oleh pentadbir

6. Apakah pandangan anda dengan sistem penyimpanan yang sedang anda gunakan?

- a. Sangat baik
- b. Baik
- c. Tidak baik
- d. Sangat tidak baik

7. Adakah anda menyokong penggunaan sistem penyimpanan berkomputer?

- a. Sangat menyokong
- b. Menyokong
- c. Tidak menyokong
- d. Sangat tidak menyokong

8. Sila jelaskan mengapa anda tidak menyokong.

9. Apakah masalah yang anda hadapi semasa menguruskan maklumat pelanggan dan kes?

- a. Carian maklumat
- b. Kerosakan maklumat
- c. Pembaziran
- d. Kebocoran maklumat
- e. Lain-lain (nyatakan) _____

**HOME
BACK
SUPPORT HOME**

Free Support Pages

[FrontPage](#)
[Sharepoint](#)
[Vizact 2000](#)
[Internet Explorer](#)
[Outlook Express](#)
[Netscape Communicator](#)
[JavaScript](#)
[Databases](#)
[Web Servers](#)
[VS.NET](#)
[ASP- Active Server Page](#)
[Internet Etiquette](#)
[Webmaster Tools](#)
[Goodies for You](#)
[Submit Free to Search Engine](#)
[Search Engine Tips](#)
[Who Links to You](#)

Web Server FAQ

Windows XP^{7/sup> Notice}

Windows XP Home Edition **DOES NOT support the Personal Web Server (PWS)** that works with previous versions of Windows. If you plan on upgrading to Windows XP and require a PWS, you need to get Windows XP Professional. For more info see MS KB Q304197. There is a work around available at BillsWay.com.

1. What is a Personal Web Server (PWS)?
2. Why is my home page renamed on the server when I publish?
3. How do I change the Default.htm page name to another name?
4. How do I change the Default.htm page name in PWS/IIS 4.0 for Windows NT?
5. How do I change the Default.htm page name in PWS 4.0 for Windows 95/98?
6. I get an Invalid Page Fault error after loading Personal Web Server 4.0, how do I fix it?
7. How do I correct the No Server on Port 80 error?
8. How can I stop the Dial-up to browse locally?
9. How do I change the URL to Publish to a different server?
10. Adding FrontPage Virtual Webs in Windows NT Internet Information Server 4.0
11. Why are the Permissions Grayed Out?
12. Why is the FTP Service no longer available?
13. Why can't I edit my local web from another machine?
14. How to I add a HOSTS file for PWS?
15. Related Microsoft Documents.
16. How do I determine which PWS version is loaded?
17. I do I fix the Transaction Server error when installing PWS?

What is a Personal Web Server (PWS)?

The Personal Web Server (PWS) is a HTTP Server that is normally installed on a development machine. It allows full testing of the web site before publishing to a server. It is also required in order to use and test many FrontPage Components. Kevin Spencer, a Microsoft MVP, has written several detailed articles for ReallyBig.com about the Personal Web Server. See the following for more information:

- [Microsoft FrontPage: Understanding the Personal Web Server](#)
- [Microsoft FrontPage: Getting Started with FrontPage98 Explorer, part 2](#)
- [Microsoft FrontPage: Web Management With FrontPage Explorer](#)

[Back to Top](#)

Why is my home page renamed on the server when I publish?

Servers running FrontPage 98 extensions automatically rename the Default Document of your web to the Default Document required by the web server. If you require a different name such as default.asp contact your service provider or systems administrator.

How do I change the Default.htm page name to another name?

The procedure to change the Default Home Page name used by FrontPage varies depending on which Personal Web Server (PWS) is installed on your computer. An easy way to tell which PWS you're using is to check in the Windows Control Panel. If there's an icon labeled Personal Web Server, you have Microsoft PWS (MS PWS) installed. If not, the chances are you have FrontPage PWS (FP PWS) installed. The other Possibility is PWS 4.0 that is part of the Option Pack for Windows NT 4.0. It includes PWS 4.0 for Windows 95.

For Personal Web Server and Internet Information Server 4.0 see:

- Windows NT Personal Web Server and Internet Information Server 4.0
- Windows 95/98 Personal Web Server 4.0

MS PWS: To change the default home page from Default.htm to Index.htm, use the following steps:

- On the Start menu, point to Settings, and then click Control Panel.
- Double-click the Personal Web Server icon.
- Click the Administration tab.
- Click Administration. This starts your default browser and loads the Administration web page.
- Click to select WWW Administration.
- Click the Directories tab.
- Select "Default Homepage."
- Rename Default.htm to Index.htm or what ever name your web server requires.
- Start FrontPage Explorer.
- On the View menu, click Folder View.
- Rename Default.htm to Index.htm or what ever name your web server requires.
- Close and then reopen the Web and your Home Page should be identified with the *Little House* symbol.

FP PWS on Windows 95 and Windows NT: To change the default home page from Default.htm to Index.htm, use the following steps:

- Using a text editor, such as Notepad or WordPad, open the **srm.cnf** file. This file is normally: **C:\FrontPage Webs\Server\Config\srm.cnf**. If this page is not located in the web, a text directory that contains Shortcuts to all existing pages will be loaded instead.
- There's a line that looks like **# DirectoryIndex index.htm**. You can remove the **#** comment, or insert a line with the name required by the web server. For example if you want your default page to be **home.html**, insert a line that looks like this:

DirectoryIndex home.html

- Save and close the **srm.cnf** file.
- Stop and restart the Personal Web Server.
- Save and close the file.
- Restart the Personal Web Server and FrontPage Explorer and new name should be recognized as the home page.

[Back to Top](#)

I get an Invalid Page Fault error after loading Personal Web Server 4.0, how do I fix it?

When you attempt to send or receive an e-mail message using Outlook Express, you may receive the following error message:

MSIMN caused an invalid page fault in module INETCOMM.DLL at <address>.

To correct the problem, see OE: ErrMsg MSIMN Caused an Invalid Page Fault..

[Back to Top](#)

How do I correct the No Server on Port 80 error?

This error is normally caused by not having the Personal Web Server (PWS) installed correctly. See the following links to get detailed instructions, which apply for both FrontPage 97 and FrontPage 98.

- Error INI File "" Section "Port 80" Not Found
- No Server on Port <Number> at <Server Name>
- "No Server on Port ..." Error Opening, Creating Web Page
- No Server on Port 80 After Laptop Computer Has Been Idle
- Personal Web Server Can't Bind to Port 80

Another cause can be a lost connection when publishing your site to the web server. Check to make sure you are connected.

[Back to Top](#)

How can I stop the Dial-up to browse locally?

In Internet Explorer use *View | Internet Options* from the pull-down menu. Go to the Connection tab and check the *Connect to internet using local area network* and click OK. You will need to use Dial-up Networking to connect manually. The easiest way is to add a Shortcut on your Desktop.

We have not tried this, however, for stopping PWS 4.0 from dialing after booting you can try Don McGrath's solution and edit the Windows Registry. Start the Registry Editor, which is normally c:\windows\regedit.exe. Locate the Key shown below.

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices

Locate and select the String **MSDTC** and right mouse click and select Modify. Edit the value from **MSDTCW -start to MSDTCW.**

NOTE: Editing the registry can render your system unusable and may require reloading the operating system and software applications.

[Back to Top](#)



How do I change the URL to Publish to a different server?

From the pull down menu in FrontPage Explorer, use File | Publish FrontPage Web | More Webs and enter the URL for the for FrontPage Web on the server. Once the uploading process is complete FrontPage assigns the new URL to the Publish button.

For more information about Publishing to a server that does not have FrontPage Extensions installed see How do I publish to a server without FrontPage Extensions?

[Back to Top](#)

Why are the Permissions Grayed Out?

Why is the FTP Service no longer available?

Why can't I edit my local web from another machine?

The problems are due to the version of the Personal Web Server and the File System on the machine. FAT16 and FAT32 file systems used by Windows 95 or Windows 98 are not secure file systems. Because of this, Microsoft disabled the Web Permissions, FTP Service and Remote Editing capabilities. Using a Unix server, Windows NT or Windows 2000 server with NTFS file systems work because the file systems are secure.

[Back to Top](#)



How to I add a HOSTS file for PWS?

1. On Windows NT, open **C:\WINNT\system32\drivers\etc\hosts** in Notepad or other text editor. On Windows 95 & Windows 98, open **C:\WINDOWS\hosts** in Notepad or other text editor. *If the hosts does not exist, use the host.sam file or create an new empty file.*
2. Add an additional line for each new wed in the format of IP Address [TAB Key] Web Name [TAB Key] [# Key] **Comments.** For example:
10.122.15.188 YourWebName # comments about your web
3. Save the file as **HOSTS** with no extension.

NOTE: You can add other servers to the list. For example adding the server where you publish your web eliminates the need to to a DNS (Domain Name Service) lookup. That saves a little time.

[Back to Top](#)



I do I fix the Transaction Server error when installing PWS 4.0?

On some Windows 95 or Windows 98 systems an error similar to the following may appear.

Error Code = 0x80004005 An unknown error occurred while making MTS specific changes to the system registry.

Followed by this error:

Setup of "Transaction Server Core Components" failed. The specific error code is 0xfee662.

To correct the problem, a updated Mtsetup.dll needs to be downloaded from Microsoft. See FILE: Mtsetup Fixes Transaction Server Core Components Setup Error on Windows 95 and Windows 98.

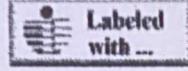
[Back to Top](#)

Related Microsoft Documents

- [How to Move FrontPage PWS to Another Drive](#)
- [Personal Web Server Can't Bind to Port 80](#)
- [How to Create an Alias for FrontPage Personal Web Server](#)
- [Why Are There Two Versions of the Personal Web Server](#)
- [When to Use FP Personal Web Server Versus Microsoft PWS](#)
- [How to Install the FrontPage Personal Web Server](#)
- [How to Install FrontPage Server Extensions for MSPWS](#)
- [Cannot Create Web Using .com, .dll, or .exe Extension](#)
- [Using FrontPage Without the Server Extensions](#)
- ["Cannot Determine Local Host Name" Error Starting PWS](#)
- [How to Change FrontPage Personal Web Server ServerName](#)
- [FrontPage Times Out While Trying to Connect to Server](#)
- [FrontPage may not Return Correct Host Name on Windows 95](#)
- [How to Use Microsoft Personal Web Server on Different Port](#)

[Back to Top](#)

For corrections, additions or comments email webmaster@SiteCrafters.com



Copyright ?2000, 2001, 2002
SiteCrafters Internet Services
All Rights Reserved

[Legal Notice about this site](#)
Last modified: March 14, 2002



Macromedia Dreamweaver UltraDev Support Center

TechNote

Defining a site that uses PWS/IIS locally and FTP remotely

This TechNote provides a sample Site definition to assist Dreamweaver UltraDev 4 users who are using Personal Web Server or a local copy of IIS for development on a Windows machine, and a remote Web server accessed via FTP for production. This Site Definition will allow you to setup one site in UltraDev to test your Web pages on your local Web server and your remote Web server. With this configuration, you do not need to use an external FTP program or make major changes to your Site Definition when switching between the local/remote Web server.

Note: For the purposes of this TechNote, the term "Web server" refers to the software that processes server side code, manages Web pages and makes them available to client computers.

Personal Web Server (PWS) can be used on Windows 98, Windows ME and Windows NT machines. Windows ME does not come with PWS and is not supported by Microsoft. Windows 2000 Professional Edition machines come with a built-in local version of IIS 5.0. This TechNote only applies to UltraDev for Windows users.

Setting up the Site Definition

- 1 From the UltraDev menu, do one of the following:
 - ▶ Select Site > Define Sites > select your site > click the Edit button.
 - ▶ Select Site > New Site.
- 2 Fill out the Local Info category as follows:

Site Definition for PWS_FTP

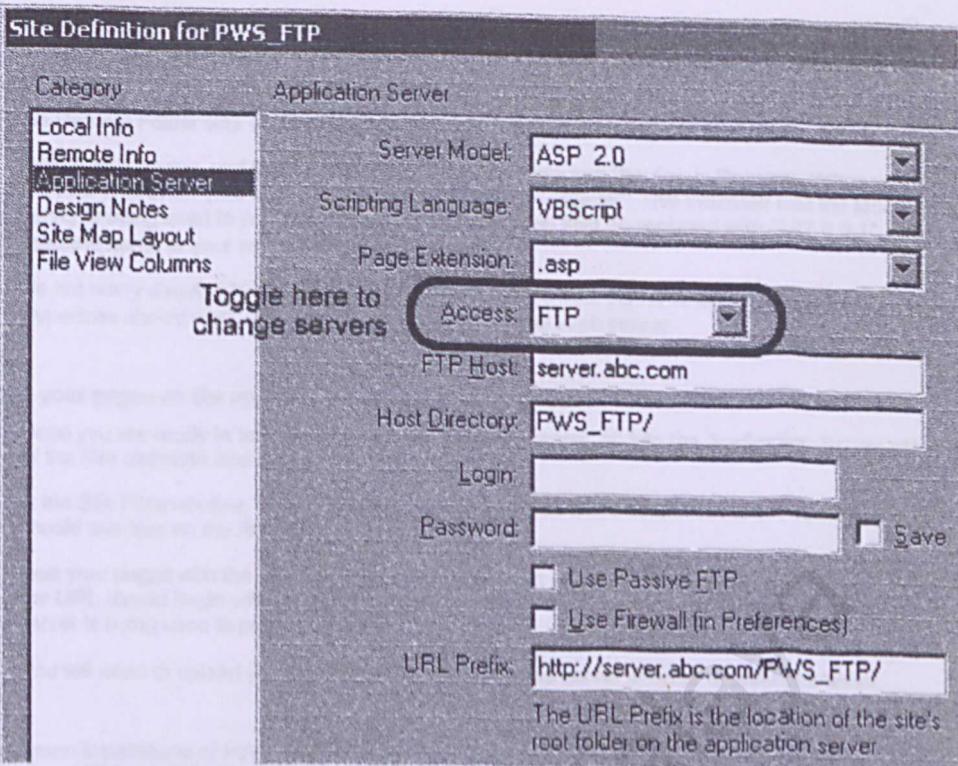
Category	Local Info
Local Info	<p>Site Name: <input type="text" value="PWS_FTP"/></p> <p>Local Root Folder: <input type="text" value="C:\Inetpub\wwwroot\PWS_FTP\"/> <input checked="" type="checkbox"/> Refresh Local File List Automatically</p>
Remote Info	
Application Server	
Design Notes	
Site Map Layout	
File View Columns	

- 3 Fill out the Remote Info category as follows:

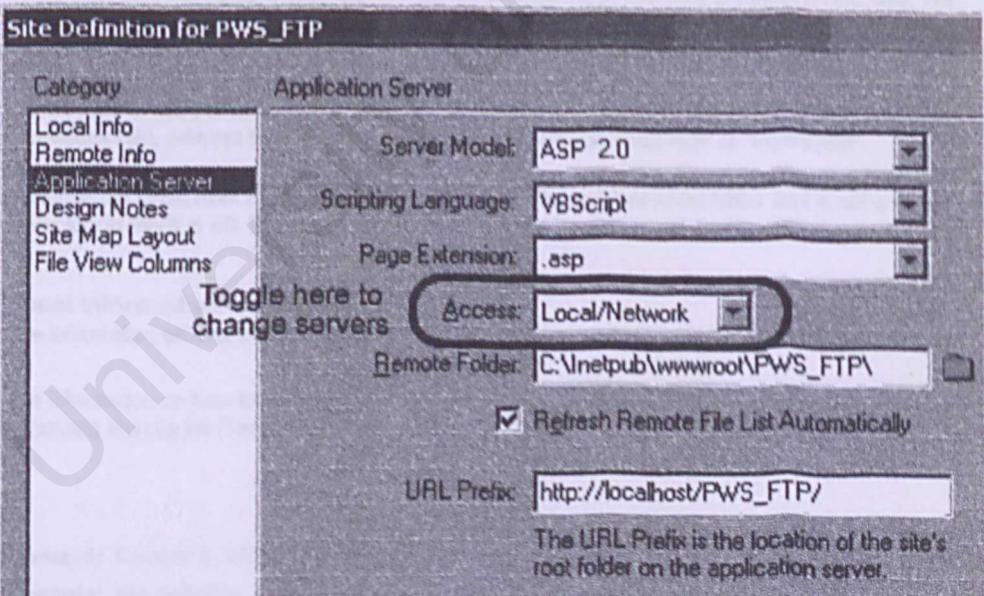
Site Definition for PWS_FTP

Category	Remote Info
Local Info	
Remote Info	<p>Access: <input type="button" value="FTP"/></p> <p>FTP Host: <input type="text" value="server.abc.com"/></p> <p>Host Directory: <input type="text" value="PWS_FTP/"/></p>
Application Server	
Design Notes	
Site Map Layout	
File View Columns	

- 4 Fill out the Application Server category appropriately. Select FTP from the Access pop-up menu. UltraDev should automatically populate the Host Directory and URL Prefix fields.



- 5 In the Application Server category, select Local/Network from the Access pop-up menu. UltraDev should automatically populate the Remote Folder and URL Prefix fields for the local Web server. You should be able to switch the Access pop-up menu back and forth between Local/Network and FTP, and UltraDev should automatically change the appropriate settings.



- 6 The site should now be configured for testing on both the local Web server and the remote FTP Web server. The Access setting will determine which Web server is used when you run File > Preview in Browser, or View > Live Data from within UltraDev.

How this Site Definition could be used

Developing your pages and testing on the local Web server:

- ▶ While developing your Web pages, set the Access to Local/Network in the Application Server category of the Site definition.
- ▶ In the Site Files window, you do not need to connect to the remote FTP host. If you do not connect, the Remote Folder side should be empty.
- ▶ Design your pages, and test them with the Live Data view and Preview in Browser. When you run Preview in Browser, the URL should begin with "http://localhost/". This indicates that the local Web server is being used to process the page. "localhost" can also be replaced with "127.0.0.1", your machine name or your machine's static IP address.
- ▶ Do not worry about uploading (or "putting") files to the remote server while in development. Hyperlinks should work correctly for the files on the local Web server.

Testing your pages on the remote Web server:

- ▶ Once you are ready to test your pages on the remote server, go into the Application Server category of the Site definition and change the Access to FTP.
- ▶ In the Site Files window, connect to the remote FTP host. Once connected to the remote server, you should see files on the Remote Folder side.

Test your pages with the Live Data view and Preview in Browser. When you run Preview in Browser, the URL should begin with "http://www.yourdomain.com/". This indicates that the remote Web server is being used to process the page.

You will need to upload (or "put") your files to the remote server in order to test hyperlinks.

Some installations of PWS and IIS, using the above configuration, will give the error "HTTP 404 - File not found". This appears to be a problem with how some installations of PWS and IIS handle file extensions. UltraDev uses temporary files when running ("TMPdroq3fvsc.htm"). These temporary files have a "hidden" file attribute in Windows. If your local Web server is using file extensions, there are some possible workarounds:

1. Set all of your Web pages .asp, .jsp or .cfm file extensions, even if they are plain HTML files. The browser will work fine without a .htm or .html extension.

2. If you want to use Preview in Browser to test your pages. Type the URL into the browser directly.

If you want to use Preview in Browser, you can replace the TMP file with the actual file name in the browser's URL address field. For example, change "TMPdroq3fvsc.htm" to "myfile.htm".

3. Set the Local Root Folder so that it is not in the C:\inetpub\wwwroot folder and is not a virtual directory. Doing so will eliminate the benefits of the configuration above.

Information

For more information on how to use PWS with UltraDev, please see [Live Data Preview and Microsoft Personal Web Server](#).

For more information on how to define a site, please see [How to define a site in Dreamweaver \(TechNote 140: Defining a Site\)](#) or [How to define a site in UltraDev \(TechNote 14619\)](#).

5, 2001

Information, PWS, 404, FTP, IIS, Personal Web Server, development, testing

1

© 2001 Macromedia, Inc. All rights reserved.

Please agree to our [Terms of Use](#).

[| Accessibility | Report Piracy](#)

- ▶ While developing your Web pages, set the Access to Local/Network in the Application Server category of the Site definition.
- ▶ In the Site Files window, you do not need to connect to the remote FTP host. If you do not connect, the Remote Folder side should be empty.
- ▶ Design your pages, and test them with the Live Data view and Preview in Browser. When you run Preview in Browser, the URL should begin with "http://localhost/". This indicates that the local Web server is being used to process the page. "localhost" can also be replaced with "127.0.0.1", your machine name or your machine's static IP address.
- ▶ Do not worry about uploading (or "putting") files to the remote server while in development. Hyperlinks should work correctly for the files on the local Web server.

Testing your pages on the remote Web server:

- ▶ Once you are ready to test your pages on the remote server, go into the Application Server category of the Site definition and change the Access to FTP.
- ▶ In the Site Files window, connect to the remote FTP host. Once connected to the remote server, you should see files on the Remote Folder side.
- ▶ Test your pages with the Live Data view and Preview in Browser. When you run Preview in Browser, the URL should begin with "http://www.yourdomain.com/". This indicates that the remote Web server is being used to process the page.
- ▶ You will need to upload (or "put") your files to the remote server in order to test hyperlinks.

Note: Some installations of PWS and IIS, using the above configuration, will give the error "HTTP 404 - File not found" in the browser on HTM or HTML files. This appears to be a problem with how some installations of PWS and IIS handle file extensions. UltraDev uses temporary files when running ("TMPdroq3fjvsc.htm"). These temporary files have a "hidden" file attribute in Windows. If your local Web server gives this error, here are some possible workarounds:

- ▶ Give all of your Web pages .asp, .jsp or .cfm file extensions, even if they are plain HTML files. The HTML pages will work fine without a .htm or .html extension.
- ▶ Do not use Preview in Browser to test your pages. Type the URL into the browser directly.
- ▶ If you want to use Preview in Browser, you can replace the TMP file with the actual file name in the browser's URL address field. For example, change "TMPdroq3fjvsc.htm" to "myfile.htm".
- ▶ Change the Local Root Folder so that it is not in the C:\inetpub\wwwroot folder and is not a virtual directory in PWS or IIS. Doing so will eliminate the benefits of the configuration above.

Additional information

For more information on how to use PWS with UltraDev, please see [Live Data Preview and Microsoft Personal Web Server](#).

For more information on how to define a site, please see [How to define a site in Dreamweaver \(TechNote 140\)](#) or [troubleshooting techniques \(TechNote 14619\)](#).

Last updated: October 5, 2001

Keywords: site definition, PWS, 404, FTP, IIS, Personal Web Server, development, testing

Created: July 3, 2001

©1995-2002 Macromedia, Inc. All rights reserved.
Use of this website signifies your agreement to the [Terms of Use](#),
[Privacy](#) | [Site Map](#) | [Contact us](#) | [Accessibility](#) | [Report Piracy](#).

Introduction to Active Server Pages

by Kevin Cooke 30 Sep 1998

Kevin Cooke is Director of Engineering at Wired Digital. He has a life-sized replica of an alien that he'd like to sell you.

Page 1

Server-side programming used to be pretty difficult. Making something work via CGI required some knowledge of arcane programming languages like Perl or C. To boot, it was inefficient. Each time someone hit a CGI script, a new process was created on the server; if your script was written in an interpreted language like Perl, the server had to start up another Perl interpreter, taking up processing time and memory. The situation got even hairier when it lived on a site that was getting a few thousand hits a day.

Also, you couldn't program in any of the nifty development interfaces, like Visual Basic or Visual C++ or Visual Anything. You were limited to watching it crash and burn, then checking out the server logs. This experience was good practice for catching stupid syntax errors, but I have since learned that I have better things to do with my life, like reading the [Ken Starr report](#).

Microsoft attempted to change all this when they introduced Active Server Pages. ASPs are server-generated pages which can call other programs to do things like [access databases](#), serve different pages to different browsers - basically, anything we used to do with CGI. ASP is almost as efficient as writing code directly to the server's application program interface, and it's a lot more efficient than CGI because it runs as a service and can take advantage of multithreaded architectures.

Here's the official word from the [Microsoft site](#): "Active Server Pages is an open, compile-free application environment in which you can combine HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. Active Server Pages enables server-side scripting for IIS with native support for both VBScript and Jscript."

Sounds great, right? But keep in mind that ASP is not a cure for cancer. True, it might heal what ails you if you're an old-school Microsoft developer who's comfortable staying in Monopoly Land. IMNSHO, ASP was introduced so that all the Visual Basic programmers in the world would feel needed again, now that the Web had shoved Unix geeks back into the limelight (they tried and failed with VBScript, which pretty much died for a whole host of reasons).

ASP stuck around, though, because it made sense. It evolved into an "open technology framework," meaning you didn't have to use Microsoft products to create code in it, though that's still the best way to go, honestly. Nowadays, you can create ASP

pages using whatever language you want, but VBScript is still the most common choice. It seems likely that more people will choose to use ASP - we do on HotBot - because all of those MS developer tools are actually pretty good and written to save you time. ASPs can also take advantage of COM and DCOM (Component Object Model and Distributed Component Object Model) objects with minimum effort.

I'll be using VBScript in my examples, but you can use JScript (JavaScript or ECMAScript or whatever) out of the box, too. Also, one company sells an out-of-the-box Perl solution for ASP, and Java is pretty easy to use with ASP, too. I'll also assume that you're using IIS 4.0 on Windows NT, even though another company has ported it to other servers and platforms. Like HTML and Perl, you can use any text editor to create ASP code. Microsoft VisualInterDev will give you nice highlights and wizards and pop-up boxes, but you may not need all that stuff. Personally, I still use vi a lot.

All set? Then let's take a look at the code.

Introduction to Active Server Pages

Page 2 — ASP Syntax

An Active Server Page is an HTML file with some additional code in it, separated from the HTML markup by a different delimiter:

```
<% ASP Code evaluated here %>
```

Here's an example page:

```
<%@ Language=VBScript %>

<HTML>
<HEAD>
<TITLE>Example 1</TITLE>
</HEAD>
<BODY bgcolor=Lime aLink=DarkTurquoise>
<P> </P>
<% Response.Write("Hello, world!") %>
</BODY>
</HTML>
```

Here are the results of this code.

If you save this file as .asp and put it on an ASP-ready server, you'll see the "Hello, world!" in the page above. If you don't, you shouldn't see anything.

The important thing to note is that the server will assume that anything between the delimiters is code and will then try to execute it. Don't put HTML there, or else it'll break. Also, the first line, <%@ Language=VBScript %> is optional. If you leave it out, the server will assume you're using VBScript.

VBScript is an odd little language. Although it supports various scalar values (integers, floating-point numbers, strings, etc.), VBScript has only one primary data type: the Variant, a peculiar data type that magically transforms itself to fit within the context in which it is currently being used. Of course, VBScript has all the requisite programming-language operators, conditional statements, and control structures.

Now that you've created one page, you're going to want to know what else is possible. The best resource, predictably, is at Microsoft. They have a complete list of available built-in ASP objects, with example code. IIS4 comes with some nice new objects, too, letting you do things like send mail and talk to ODBC-based databases. Here's the scoop on these objects from the [Microsoft site](#):

- **Request** - to get information from the user
- **Response** - to send information to the user
- **Server** - to control the Internet Information Server
- **Session** - to store information about and change settings for the user's current Web-server session
- **Application** - to share application-level information and control settings for the lifetime of the application

Each of these objects has a collection of functions that it can perform, called "methods," and one or more properties, each called with the usual object-orientation-style code [value = Object.Property or value = Object.Method()]. The "Hello, world!" example above uses one of the most common methods of the Response object: write. You can create objects that are ASP-accessible by using the Component Object Model, Microsoft's new way of creating OO code. If you need an intro to COM or need to know what I mean by properties and methods, you can find good definitions [here](#).

If you want to make your own widgets, you can either write your own COM objects or you can buy them from a variety of companies. It's likely that there is already an object out there that does what you need.

Now that you have a basic understanding of ASP and its related objects, let's try something a little more practical: browser detection.

Introduction to Active Server Pages

Page 3 — Browser Detection

You can obviously use client-side scripting for browser detection, but if the client doesn't support client-side scripting, like Lynx, or people who are paranoid and turn JavaScript off,

you're hosed.

Take a look at this example for an idea of how ASP can handle this problem:

```
<%@ Language=VBScript %>

<HTML>

<HEAD>

<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">

</HEAD>

<BODY>

<%
dim BrowserType

set bc = Server.CreateObject("MSWC.BrowserType")

if bc.browser="IE" then
    BrowserType = "MSIE"
elseif bc.browser="Netscape" then
    BrowserType = "Netscape"
elseif bc.browser="Lynx" then
    BrowserType = "Lynx"
end if

%>

<%
select case BrowserType
case "Lynx"
    Response.Write("You're using Lynx! How do you manage to live
without 3-D backgrounds and endless download times? Whatever!")
case "MSIE"
    Response.Write("You're using Internet Explorer! Thank you for
helping keep Microsoft afloat!")
case "Netscape"

```

```
Response.Write("You're using Netscape! And you're wearing those  
great pink pants! ASP knows everything about you!")
```

```
case else
```

```
Response.Write("You're using some other browser I don't know  
about.")
```

```
End select
```

```
%>
```

```
<BR>
```

```
</BODY>
```

```
</HTML>
```

Take a look at the results of this code. It should tell you what type of browser you're running.

The folks at Microsoft have provided lots of objects, like the MSWC.BrowserType above, which can make your life easier. This server object's property, MSWC.BrowserType.browser, is much simpler to deal with than the usual HTTP_USER_AGENT because it doesn't say "Mozilla (compatible)" for Internet Explorer, and you don't have to parse the string just to get the version number. It's another property, (majorver), as is the client platform (platform). It just makes for cleaner code, which is always a good thing. With a few more branches in your if statement, you could serve up a page for all of the major browser, platform, and version groups to be absolutely sure that users get the best browsing experience possible. We use something similar at HotBot for the above script to server-side-include different ASP files for different browsers. Try using HotBot with Lynx - it's groovy!

```
[code snippet: If bc.majorver = "4" and InStr(bc.platform,"Win") <> 0  
then ]
```

It would be a good idea to store the platform information on a cookie, so you don't have to do all of the server-side processing with every request for this page, but only if you're really concerned about traffic on your site.

In sum, ASP is pretty powerful stuff, and it's fairly easy. Even if you personally don't want to contribute to Microsoft's growing power, it might make sense for you to take a look at what the folks in Redmond have come up with. If you'd like to see a bit more of what ASP can do, take a look at Jay's database tutorial.

Seeing a Lawyer

(As at March 1997)

Lawyers

When you wish to see a lawyers

Services offered by Lawyers

How to look for lawyers

Seeing a lawyer

Legal fees

Legal aid

Bar council

LAWYERS

Lawyers are professional people trained in the law. You may seek their help to solve your problems. They are independent people and any information given to them is confidential and can only be released with your consent. Though a lawyer's main duty is to protect your interest, he is also an officer of the Court and must not mislead the court or assist you in any way to break the law. He is bound by a strict code of conduct and rules of etiquette.

WHEN YOU MAY WISH TO SEE A LAWYER

Generally you may wish to see a lawyer to:

- Get advice on a legal problem - Represent you in court, if necessary - Assist you in the handling of your legal rights

You may therefore wish to see lawyer, in particular:

- When you are charged with a criminal offence, to seek advice on matters relating to your defence, and to represent you in court.
- When you wish to make a claim against another party you can ask a lawyer to file a claim for you and to represent you in court.
- When some other party has unfairly filed a claim against you in Court, you may wish to seek the help of a lawyer to defend the claim.

When you are in doubt over any matter relating to the law, it is advisable to consult a lawyer. This is for your own protection. Also, taking legal advice at an early stage could save you a lot of time, trouble and money.

SERVICES OFFERED BY LAWYERS

Lawyers can provide you with a variety of important and useful services. Some of the services provided by them are:

- When somebody owes you money and has not repaid it you despite repeated requests, you may wish to see a lawyer to make a claim your behalf.
- When you are involved in a traffic accident, a lawyer can advise you on the steps to be taken in making a claim, as well as what you should do to protect yourself against claims from other parties.
- When you have a problem with your husband or wife, a lawyer can advise you on the law relating to divorce, separation, maintenance and custody of children.

- When you wish to buy or sell a house, a lawyer can advise you on the procedures involved, check all documents on yourself.
- When you have a problem relating to tenancy of a property, either as landlord or tenant, you can see a lawyer to seek advice or represent you in Court, if necessary.
- Before you sign any important, business or employment contract, you could show a draft of it to a lawyer and seek his advice.
- When a family member has passed away, you can ask a lawyer to apply to court to enable the assets of the deceased to be dealt with.

You may wish to consult a lawyer:

- To make a will so that your assets will be distributed according to your wishes after your death
- To change your name
- To adopt a child
- Before you start a business or incorporate a company
- To find out your legal rights in the matter.

HOW TO LOOK FOR LAWYERS

Lawyers are not allowed to advertise but the Bar Council publishes a directory of lawyers which contains names, addresses and other relevant information of all practicing lawyers in Malaysia. The best method of choosing a lawyer is by personal recommendation of friends, colleagues and relatives.

The names, addresses and telephone numbers of lawyers and their firms can be found in the telephone directory.

SEEING A LAWYER

If your first contact with your lawyer is by telephone, inform him of your reason for consulting him and then make an appointment to see him.

At the meeting, bring along with you all relevant documents and give your lawyer all the necessary information. It would be helpful to write down all the questions you would like to ask your lawyer.

Do not be afraid to ask questions. If you wish, you may also ask your lawyer about the amount of work involved and how long your matter would take.

Remember: Your lawyer is there to assist you.

LEGAL FEES

The fees payable by you will generally depend on the amount of work done on your behalf, the time taken and the difficulty of the work involved. In some cases, the fees payable by you are fixed by law e.g. the buying and selling of houses.

It is common for lawyers to ask you to place a deposit with them before they begin work on your behalf. You can request for a receipt for any money placed by you with a lawyer.

Always feel free to discuss the question of fees with your lawyer. This is to avoid misunderstanding. If you are unhappy with the fees charged to you, discuss the matter with your lawyer first. If the matter cannot be resolved, you may ask your lawyer to arrange for his bill to be taxed i.e. decided by the court.

LEGAL AID

If there is a need for you to see a lawyer but you cannot afford to pay his fees, you may seek help from a Legal Aid Centre run by the Bar Council. The Legal Aid Centre in Kuala Lumpur is situated at:

Tingkat 5, Bangunan Straits Trading,
4, Lebuh Pasar Besar,
50050 Kuala Lumpur.
Tel: 2913005/2932072

Legal Aid Centres are located in most cities and towns in Peninsular Malaysia.

The Centres provide representation for all criminal matters for all those who qualify except offences that carry capital punishment (i.e. death and life sentence) and render legal assistance for all civil matters except matters relating to recovery of debts and motor accident claims.

The Government also runs a Legal Aid Scheme for people who need legal assistance but cannot afford to pay fees. The Government Legal Aid Bureau has offices in all major cities and towns in Malaysia and their office at Kuala Lumpur is situated at:

Biro Bantuan Guaman,
Wilayah Persekutuan,
Tingkat 1, Wisma Cycle & Carriage,
50504 Kuala Lumpur.
Tel :2942700

The Legal Aid Bureau renders assistance in most legal matters for those who qualify except legal representation for criminal cases.

It should be noted that a person seeking legal aid has to pass a strict Means Test to qualify" for assistance from either the Bar Council Legal Aid Centre or the Government Legal Aid Bureau. You will be required to give information on your income, savings

and property owned.

BAR COUNCIL

BAR COUNCIL The Bar Council is the governing body for lawyers practising in Peninsular Malaysia. It sets the standard of professional conduct which lawyers have to follow. If you are unhappy with the services provided by your lawyer, you should first try to sort it out with him, as the problem may be the result of a misunderstanding or a breakdown in communication. If this is not possible you may wish to write to the Bar Council for assistance.

The address of the Bar Council is:

Bar Council,
Jalan Tun Perak,
50050 Kuala Lumpur
Tel: 2911312/1366/1367/1512/1689
Fax: (03) 2912439
E-Mail: [cateu @ bar.po.my.](mailto:cateu@bar.po.my)

(Reproduced with the courtesy of the Bar Council.)

Copyright © 2002 PNMB-LawNet. All rights reserved.

Law *and later Successor to British, William the Conqueror died in 1066.*
All the rules of conduct that have been approved by the government and which are in force over a certain territory and which must be obeyed by all persons on that territory (eg. the "laws" of Australia). Violation of these rules could lead to government action such as imprisonment or fine, or private action such as a legal judgement against the offender obtained by the person injured by the action prohibited by law. Synonymous to act or statute although in common usage, "law" refers not only to legislation or statutes but also to the body of unwritten law in those states which recognize common law.

Justice

Fairness. A state of affairs in which conduct or action is both fair and right, given the circumstances. In law, it more specifically refers to the paramount obligation to ensure that all persons are treated fairly. Litigants "seek justice" by asking for compensation for wrongs committed against them; to right the inequity such that, with the compensation, a wrong has been righted and the balance of "good" or "virtue" over "wrong" or "evil" has been corrected.

Case law

The entire collection of published legal decisions of the courts which, because of stare decisis, contributes a large part of the legal rules which apply in modern society. If a rule of law cannot be found in written laws, lawyers will often say that it is a rule to be found in "case law". In other words, the rule is not in the statute books but can be found as a principle of law established by a judge in some recorded case. The word jurisprudence has become synonymous for case law.

Lawyer

A person that has been trained in the law and that has been certified to give legal advice or to represent others in litigation. Also known as a "barrister & solicitor" or an attorney.

law

n. 1) any system of regulations to govern the conduct of the people of a community, society or nation, in response to the need for regularity, consistency and justice based upon collective human experience. Custom or conduct governed by the force of the local king were replaced by laws almost as soon as man learned to write. The earliest lawbook was written about 2100 B.C. for Ur-Nammu, king of Ur, a Middle Eastern city-state. Within three centuries Hammurabi, king of Babylonia, had enumerated laws of private conduct, business and legal precedents, of which 282 articles have survived. The term "eye for an eye" (or the equivalent value) is found there, as is drowning as punishment for adultery by a wife (while a husband could have slave concubines), and unequal treatment of the rich and the poor was codified here first. It took another thousand years before written law codes developed among the Greek city-states (particularly Athens) and Israel. China developed similar rules of conduct, as did Egypt. The first law system which has a direct influence on the American legal system was the codification of all classic law ordered by the Roman Emperor Justinian in 528 and completed by 534, becoming the law of the Roman empire. This is known as the Justinian Code, upon which most of the legal systems of most European nations are based to this day. The principal source of American law is the common law, which had its roots about the same time as Justinian, among Angles,

Britons and later Saxons in Britain. William the Conqueror arrived in 1066 and combined the best of this Anglo-Saxon law with Norman law, which resulted in the English common law, much of which was by custom and precedent rather than by written code. The American colonies followed the English Common Law with minor variations, and the four-volume Commentaries on the Laws of England by Sir William Blackstone (completed in 1769) was the legal "bible" for all American frontier lawyers and influenced the development of state codes of law. To a great extent common law has been replaced by written statutes, and a gigantic body of such statutes have been enacted by federal and state legislatures supposedly in response to the greater complexity of modern life. 2) n. a statute, ordinance or regulation enacted by the legislative branch of a government and signed into law, or in some nations created by decree without any democratic process. This is distinguished from "natural law," which is not based on statute, but on alleged common understanding of what is right and proper (often based on moral and religious precepts as well as common understanding of fairness and justice). 3) n. a generic term for any body of regulations for conduct, including specialized rules (military law), moral conduct under various religions and for organizations, usually called "bylaws."

University of Malaya

PENGKODAN

University of Malaya

UNTUK PENGHAPUSAN MAKLUMAT DARI PENGKALAN DATA

```
<%@LANGUAGE="VBSCRIPT"%>
<!--#include file="Connections/deleteFile.asp" -->
<%
' *** Restrict Access To Page: Grant or deny access to this page
MM_authorizedUsers="admin,Lawyer,ADMIN,Admin"
MM_authFailedURL="unauthorizedMessage.asp"
MM_grantAccess=false
If Session("MM_Username") <> "" Then
  If (false Or CStr(Session("MM_UserAuthorization"))="" ) Or _
    (InStr(1,MM_authorizedUsers,Session("MM_UserAuthorization"))>=1) Then
    MM_grantAccess = true
  End If
End If
If Not MM_grantAccess Then
  MM_qsChar = "?"
  If (InStr(1,MM_authFailedURL,"?") >= 1) Then MM_qsChar = "&"
  MM_referrer = Request.ServerVariables("URL")
  if (Len(Request.QueryString()) > 0) Then MM_referrer = MM_referrer & "?" & Request.QueryString()
  MM_authFailedURL = MM_authFailedURL & MM_qsChar & "accessdenied=" &
  Server.URLEncode(MM_referrer)
  Response.Redirect(MM_authFailedURL)
End If
%>
<%
' *** Edit Operations: declare variables

MM_editAction = CStr(Request("URL"))
If (Request.QueryString <> "") Then
  MM_editAction = MM_editAction & "?" & Request.QueryString
End If

' boolean to abort record edit
MM_abortEdit = false

' query string to execute
MM_editQuery = ""
%>
<%
' *** Delete Record: declare variables

if (CStr(Request("MM_delete")) <> "" And CStr(Request("MM_recordId")) <> "") Then

  MM_editConnection = MM_deleteFile_STRING
  MM_editTable = "insertData"
  MM_editColumn = "ID"
  MM_recordId = "" + Request.Form("MM_recordId") + ""
  MM_editRedirectUrl = "berjayainsert.asp"

  ' append the query string to the redirect URL
  If (MM_editRedirectUrl <> "" And Request.QueryString <> "") Then
    If (InStr(1, MM_editRedirectUrl, "?", vbTextCompare) = 0 And Request.QueryString <> "") Then
      MM_editRedirectUrl = MM_editRedirectUrl & "?" & Request.QueryString
    Else
      MM_editRedirectUrl = MM_editRedirectUrl & "&" & Request.QueryString
    End If
  End If

  End If
%>
<%
' *** Delete Record: construct a sql delete statement and execute it

If (CStr(Request("MM_delete")) <> "" And CStr(Request("MM_recordId")) <> "") Then
```

```

' create the sql delete statement
MM_editQuery = "delete from " & MM_editTable & " where " & MM_editColumn & " = " & MM_recordId

If (Not MM_abortEdit) Then
    ' execute the delete
    Set MM_editCmd = Server.CreateObject("ADODB.Command")
    MM_editCmd.ActiveConnection = MM_editConnection
    MM_editCmd.CommandText = MM_editQuery
    MM_editCmd.Execute
    MM_editCmd.ActiveConnection.Close

    If (MM_editRedirectUrl <> "") Then
        Response.Redirect(MM_editRedirectUrl)
    End If
End If

End If
%>
<%
Dim deleteFile__MMColParam
deleteFile__MMColParam = "1"
if (Request.QueryString("ID") <> "") then deleteFile__MMColParam = Request.QueryString("ID")
%>
<%
set deleteFile = Server.CreateObject("ADODB.Recordset")
deleteFile.ActiveConnection = MM_deleteFile_STRING
deleteFile.Source = "SELECT * FROM insertData WHERE ID = " + Replace(deleteFile__MMColParam, "", "")
deleteFile.CursorType = 0
deleteFile.CursorLocation = 2
deleteFile.LockType = 3
deleteFile.Open()
deleteFile_numRows = 0
%>
<html>
<head>
<title>delete</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000" background="images/background.gif">
<p align="center"><font face="Verdana, Arial, Helvetica, sans-serif" size="3" color="#000033"><a href="displayFile.asp?ID=<%=(deleteFile.Fields.Item("ID").Value)%>"><b>BACK</b></a><b>
</b></font><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000033"><b><font size="3"><a href="search_record_byFile.asp">MAIN</a></font></b></font><a href="displayFile.asp"><b></b></a></p>
<form ACTION="<%="MM_editAction%">" METHOD="POST" name="form1">
<table width="100%" border="0" bgcolor="#000066">
<tr>
<td>
<div align="center"><font color="#FFFFFF"><b><font size="5" face="Verdana, Arial, Helvetica, sans-serif">
CLIENT PARTICULAR</font></b></font></div>
</td>
</tr>
</table>
<p align="center">
<input type="submit" name="Submit" value="Delete record">
</p>
<p align="right"><a href="http://localhost/norzita/form_deletedataFile.asp#bottom">
</a></p>
<table width="100%" border="0">
<tr>
<td width="30%"><b><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">File
Reference </font></b></td>
<td width="70%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#990000"><b><font color="#006600"><%=(deleteFile.Fields.Item("FILE_REFERENCE").Value)%></font> </b></font></td>
</tr>

```

```

<tr>
<td width="30%"><b><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">Status
Of Case</font></b></td>
<td width="70%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000"><%=deleteFile.Fields.Item("STATUS_CASE").Value%> </font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif"><b>Client
Particular</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Name</font></td>
<td width="70%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=deleteFile.Fields.Item("NAME_CLIENT").Value%> </font></td>
</tr>
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Identification
Number</font></td>
<td width="70%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=deleteFile.Fields.Item("IC_CLIENT").Value%> </font></td>
</tr>
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Address</font></td>
<td width="70%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=deleteFile.Fields.Item("ADDRESS_CLIENT").Value%> </font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif">The
Other Partie's Particular</font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#000066">Name</font></td>
<td width="70%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-
serif"><%=deleteFile.Fields.Item("NAME_PARTY").Value%> </font></td>
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#000066">Identification
Number</font></td>
<td width="70%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-
serif"><%=deleteFile.Fields.Item("IC_PARTY").Value%> </font></td>
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#000066">Address</font></td>
<td width="70%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-
serif"><%=deleteFile.Fields.Item("ADDRESS_PARTY").Value%> </font></td>
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">Solicitor
Name and Address</font></td>
<td width="70%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-
serif"><%=deleteFile.Fields.Item("SOLICITOR_ADDRESS").Value%> </font></td>
</tr>
</table>

```

```

<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#FFFFFF"><b>Particular
In Certificate Of Marriage</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">Date</font></td>
<td width="61%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=(deleteFile.Fields.Item("DATE_CERT").Value)%> </font></td>
</tr>
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">State</font></td>
<td width="61%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=(deleteFile.Fields.Item("STATE_CERT").Value)%> </font></td>
</tr>
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-serif">Address
of Registration Office</font></td>
<td width="61%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=(deleteFile.Fields.Item("ADDRESS_CERT").Value)%> </font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><b><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif">Ground
Of Divorced</font></b></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#990000"><%=(deleteFile.Fields.Item("GROUND_DIVORCED").Value)%> </font></td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td>
<div align="right"> </div>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font size="2" color="#FFFFFF"><b><font face="Verdana, Arial, Helvetica, sans-serif">Child
Name</font></b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">1.</font></td>
<td width="94%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000"><%=(deleteFile.Fields.Item("CHILD1").Value)%> </font></td>
</tr>
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">2.</font></td>
<td width="94%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000"><%=(deleteFile.Fields.Item("CHILD2").Value)%> </font></td>
</tr>
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">3.</font></td>

```

```

<td width="94%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD3").Value)%</font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FFFFFF"><b>Particular
In Property (if any)</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="6%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">1.</font></td>
<td width="94%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("PROPERTY1").Value)%</font></td>
</tr>
<tr>
<td width="6%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">2.</font></td>
<td width="94%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("PROPERTY2").Value)%</font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FFFFFF"><b>Client
Claimed Against The Other Parties</b></font></td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#FF0000"><b>Children</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Age</font></td>
<td width="78%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD_AGE").Value)%</font></td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Name</font></td>
<td width="78%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD_NAME").Value)%</font></td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Date
Of Birth</font></td>
<td width="78%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD_DATE").Value)%</font></td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Place
Of Birth</font></td>
<td width="78%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD_PLACE").Value)%</font></td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Sex</font></td>
<td width="78%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#990000">%=(deleteFile.Fields.Item("CHILD_SEX").Value)%</font></td>
</tr>

```

```

<table>
<table width="100%" border="0">
<tr>
<td><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#FF0000"><b>Property</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Type
    Of Property 1</font></td>
<td width="72%"><font color="#990000" face="Verdana, Arial, Helvetica, sans-serif" size="2"><%=(deleteFile.Fields.Item("TYPE1").Value)%> </font></td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Location</font></td>
<td width="72%"><font color="#990000" face="Verdana, Arial, Helvetica, sans-serif" size="2"><%=(deleteFile.Fields.Item("LOCATION1").Value)%> </font></td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Assess</font></td>
<td width="72%"><font size="2" color="#990000" face="Verdana, Arial, Helvetica, sans-serif">RM
    <%=(deleteFile.Fields.Item("ASSESS1").Value)%> </font> </td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Type
    Of Property 2</font></td>
<td width="72%"><font color="#990000" face="Verdana, Arial, Helvetica, sans-serif" size="2"><%=(deleteFile.Fields.Item("TYPE2").Value)%> </font></td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Location</font></td>
<td width="72%"><font color="#990000" face="Verdana, Arial, Helvetica, sans-serif" size="2"><%=(deleteFile.Fields.Item("LOCATION2").Value)%> </font></td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Assess</font></td>
<td width="72%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#990000">RM
    <%=(deleteFile.Fields.Item("ASSESS2").Value)%> </font> </td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FF0000"><b>Maintenance</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="33%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">How
    much to proposed</font></td>
<td width="67%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-serif">RM
    <%=(deleteFile.Fields.Item("PROPOSED").Value)%> </font></td>
</tr>
<tr>
<td width="33%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">How
    much court decided</font></td>
<td width="67%"><font color="#990000" size="2" face="Verdana, Arial, Helvetica, sans-serif">RM
    <%=(deleteFile.Fields.Item("DECIDED").Value)%> </font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>

```

```

<td><font size="2" color="#FFFFFF" face="Verdana, Arial, Helvetica, sans-serif"><b>Invoice</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="41%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Paid</font></td>
<td width="59%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#990000">RM
<%=(deleteFile.Fields.Item("PAID").Value)%> </font></td>
</tr>
<tr>
<td width="41%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Any
outstanding amount to be charged</font></td>
<td width="59%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#990000">RM
<%=(deleteFile.Fields.Item("BALANCE").Value)%> </font></td>
</tr>
</table>
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p align="right">&nbsp;</p>
<p align="right">&nbsp;</p>
<p>&nbsp;</p>
<input type="hidden" name="MM_delete" value="true">
<input type="hidden" name="MM_recordId" value=<%= deleteFile.Fields.Item("ID").Value %>>
</form>
</body>
</html>
<%
deleteFile.Close()
%>

```

UNTUK MENGEMASKINI MAKLUMAT PENGKALAN DATA

```

<%@LANGUAGE="VBSCRIPT"%>
<!--#include file="Connections/modify.asp" -->
<%
' *** Restrict Access To Page: Grant or deny access to this page
MM_authorizedUsers="admin, Lawyer, ADMIN, Admin"
MM_authFailedURL="unauthorizedMessage.asp"
MM_grantAccess=false
If Session("MM_Username") <> "" Then
If (false Or CStr(Session("MM_UserAuthorization"))="") Or _
(InStr(1,MM_authorizedUsers,Session("MM_UserAuthorization"))>=1) Then
    MM_grantAccess = true
End If
End If
If Not MM_grantAccess Then
    MM_qsChar = "?"
    If (InStr(1,MM_authFailedURL,"?") >= 1) Then MM_qsChar = "&"
    MM_referrer = Request.ServerVariables("URL")
    if (Len(Request.QueryString()) > 0) Then MM_referrer = MM_referrer & "?" & Request.QueryString()
    MM_authFailedURL = MM_authFailedURL & MM_qsChar & "accessdenied=" &
Server.URLEncode(MM_referrer)
    Response.Redirect(MM_authFailedURL)
End If
%>
<%
' *** Edit Operations: declare variables
MM_editAction = CStr(Request("URL"))
If (Request.QueryString <> "") Then
    MM_editAction = MM_editAction & "?" & Request.QueryString
End If
' boolean to abort record edit
MM_abortEdit = false

```

```

' query string to execute
MM_editQuery = ""
%>
<%
' *** Update Record: set variables

If (CStr(Request("MM_update")) <> "" And CStr(Request("MM_recordId")) <> "") Then

    MM_editConnection = MM_modify_STRING
    MM_editTable = "insertData"
    MM_editColumn = "ID"
    MM_recordId = "" + Request.Form("MM_recordId") + ""
    MM_editRedirectUrl = "berjayainsert.asp"
    MM_fieldsStr =
"fileReference|value|textfield|value|clientname|value|clientIC|value|clientAddress|value|partyName|value|ICparty|
value|partyAddress|value|solicitorAddress|value|datecertificate|value|statecertificate|value|addressRegistration|val
ue|ground|value|child1|value|child2|value|child3|value|particular1|value|particular2|value|ageclaim|value|nameclai
m|value|dateclaim|value|placeclaim|value|selectsex|value|type1|value|location1|value|assess1|value|type2|value|loc
ation2|value|assess2|value|proposed|value|decided|value|paid|value|baki|value"

    MM_columnsStr =
"FILE_REFERENCE|none, "STATUS_CASE|none, "NAME_CLIENT|none, "IC_CLIENT|none, none, NULL|
ADDRESS_CLIENT|none, "NAME_PARTY|none, "IC_PARTY|none, none, NULL|ADDRESS_PARTY|none, |
SOLICITOR_ADDRESS|none, "DATE_CERT|none, "STATE_CERT|none, "ADDRESS_CERT|none, "GR
OUND_DIVORCED|none, "CHILD1|none, "CHILD2|none, "CHILD3|none, "PROPERTY1|none, "PROPER
TY2|none, "CHILD_AGE|none, none, NULL|CHILD_NAME|none, "CHILD_DATE|none, "CHILD_PLACE|n
one, "CHILD_SEX|none, "TYPE1|none, "LOCATION1|none, "ASSESS1|none, none, NULL|TYPE2|none, "L
OCATION2|none, "ASSESS2|none, none, NULL|PROPOSED|none, none, NULL|DECIDED|none, none, NULL|PA
ID|none, none, NULL|BALANCE|none, none, NULL"

    ' create the MM_fields and MM_columns arrays
    MM_fields = Split(MM_fieldsStr, "|")
    MM_columns = Split(MM_columnsStr, "|")

    ' set the form values
    For i = LBound(MM_fields) To UBound(MM_fields) Step 2
        MM_fields(i+1) = CStr(Request.Form(MM_fields(i)))
    Next

    ' append the query string to the redirect URL
    If (MM_editRedirectUrl <> "" And Request.QueryString <> "") Then
        If (InStr(1, MM_editRedirectUrl, "?", vbTextCompare) = 0 And Request.QueryString <> "") Then
            MM_editRedirectUrl = MM_editRedirectUrl & "?" & Request.QueryString
        Else
            MM_editRedirectUrl = MM_editRedirectUrl & "&" & Request.QueryString
        End If
    End If

    End If
%>
<%
' *** Update Record: construct a sql update statement and execute it

If (CStr(Request("MM_update")) <> "" And CStr(Request("MM_recordId")) <> "") Then

    ' create the sql update statement
    MM_editQuery = "update " & MM_editTable & " set "
    For i = LBound(MM_fields) To UBound(MM_fields) Step 2
        FormVal = MM_fields(i+1)
        MM_typeArray = Split(MM_columns(i+1), ",")
        Delim = MM_typeArray(0)
        If (Delim = "none") Then Delim = ""
        AltVal = MM_typeArray(1)
        If (AltVal = "none") Then AltVal = ""
        EmptyVal = MM_typeArray(2)
        If (EmptyVal = "none") Then EmptyVal = ""
        If (FormVal = "") Then

```

```

FormVal = EmptyVal
Else
If (AltVal <> "") Then
    FormVal = AltVal
ElseIf (Delim = "") Then ' escape quotes
    FormVal = "" & Replace(FormVal, "", "") & ""
Else
    FormVal = Delim + FormVal + Delim
End If
End If
If (i <> LBound(MM_fields)) Then
    MM_editQuery = MM_editQuery & ","
End If
MM_editQuery = MM_editQuery & MM_columns(i) & " = " & FormVal
Next
MM_editQuery = MM_editQuery & " where " & MM_editColumn & " = " & MM_recordId

If (Not MM_abortEdit) Then
    ' execute the update
    Set MM_editCmd = Server.CreateObject("ADODB.Command")
    MM_editCmd.ActiveConnection = MM_editConnection
    MM_editCmd.CommandText = MM_editQuery
    MM_editCmd.Execute
    MM_editCmd.ActiveConnection.Close

    If (MM_editRedirectUrl <> "") Then
        Response.Redirect(MM_editRedirectUrl)
    End If
End If

End If
%>
<%>
Dim modify__MMColParam
modify__MMColParam = "1"
if (Request.QueryString("ID") <> "") then modify__MMColParam = Request.QueryString("ID")
%>
<%>
set modify = Server.CreateObject("ADODB.Recordset")
modify.ActiveConnection = MM_modify_STRING
modify.Source = "SELECT * FROM insertData WHERE ID = " + Replace(modify__MMColParam, "", "") + ""
modify.CursorType = 0
modify.CursorLocation = 2
modify.LockType = 3
modify.Open()
modify_numRows = 0
%>
<html>
<head>
<title>modify</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000" background="images/background.gif">
<div align="center"><font size="3" face="Verdana, Arial, Helvetica, sans-serif" color="#000066"><b><font
color="#000033"><a href="search_record_byFile.asp">MAIN</a></font></b></font>
<a href="displayFile.asp"><b><font size="3" face="Verdana, Arial, Helvetica, sans-serif">BACK
</font></b></a></div>
<form ACTION="<%>MM_editAction%>" METHOD="POST" name="form1">
<table width="100%" border="0" bgcolor="#000066">
<tr>
<td>
<div align="center"><font color="#FFFFFF"><b><font size="5" face="Verdana, Arial, Helvetica, sans-
serif">CLIENT
    PARTICULAR</font></b></font></div>
</td>
</tr>
</table>

```

```

<p align="center">
<input type="submit" name="Submit2" value="Update record">
</p>
<p align="right"><a href="http://localhost/norzita/modifyRecord.asp#bottom">
</a></p>
<table width="100%" border="0">
<tr>
<td width="30%"><b><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">File
Reference </font></b></td>
<td width="70%">
<input type="text" name="fileReference"
value="<%=(modify.Fields.Item("FILE_REFERENCE").Value)%>">
</td>
</tr>
<tr>
<td width="30%"><b><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">Status
Of Case</font></b></td>
<td width="70%">
<textarea name="textfield" rows="4"
cols="50"><%=(modify.Fields.Item("STATUS_CASE").Value)%></textarea>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif"><b>Client
Particular</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Name</font></td>
<td width="70%">
<input type="text" name="clientname" value="<%=(modify.Fields.Item("NAME_CLIENT").Value)%>">
</td>
</tr>
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Identification
Number</font></td>
<td width="70%">
<input type="text" name="clientIC" value="<%=(modify.Fields.Item("IC_CLIENT").Value)%>">
</td>
</tr>
<tr>
<td width="30%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-
serif">Address</font></td>
<td width="70%">
<textarea name="clientAddress" cols="50"
rows="4"><%=(modify.Fields.Item("ADDRESS_CLIENT").Value)%></textarea>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif">The
Other Partie's Particular</font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif"
color="#000066">Name</font></td>
<td width="70%">
<input type="text" name="partyName" value="<%=(modify.Fields.Item("NAME_PARTY").Value)%>">
</td>

```

```
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif">
color="#000066">Identification
    Number</font></td>
<td width="70%">
    <input type="text" name="ICparty" value="<%=(modify.Fields.Item("IC_PARTY").Value)%>">
</td>
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif">
color="#000066">Address</font></td>
<td width="70%">
    <textarea name="partyAddress" rows="4"
cols="50"><%=(modify.Fields.Item("ADDRESS_PARTY").Value)%></textarea>
</td>
</tr>
<tr>
<td width="30%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">Solicitor
Name and Address</font></td>
<td width="70%">
    <textarea name="solicitorAddress" cols="50"
rows="5"><%=(modify.Fields.Item("SOLICITOR_ADDRESS").Value)%></textarea>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#FFFFFF"><b>Particular
In Certificate Of Marriage</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">Date</font></td>
<td width="61%">
    <input type="text" name="datecertificate" value="<%=(modify.Fields.Item("DATE_CERT").Value)%>">
</td>
</tr>
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">State</font></td>
<td width="61%">
    <input type="text" name="statecertificate" value="<%=(modify.Fields.Item("STATE_CERT").Value)%>">
</td>
</tr>
<tr>
<td width="39%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-serif">Address
of Registration Office</font></td>
<td width="61%">
    <textarea name="addressRegistration" cols="50"
rows="4"><%=(modify.Fields.Item("ADDRESS_CERT").Value)%></textarea>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><b><font color="#FFFFFF" size="2" face="Verdana, Arial, Helvetica, sans-serif">Ground
Of Divorced</font></b></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td>
    <textarea name="ground" cols="80"
rows="7"><%=(modify.Fields.Item("GROUND_DIVORCED").Value)%></textarea>

```

```

</td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td>
<div align="right"> </div>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font size="2" color="#FFFFFF"><b><font face="Verdana, Arial, Helvetica, sans-serif">Child
Name</font></b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">1.</font></td>
<td width="94%">
<input type="text" name="child1" value="<%=(modify.Fields.Item("CHILD1").Value)%>">
</td>
</tr>
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">2.</font></td>
<td width="94%">
<input type="text" name="child2" value="<%=(modify.Fields.Item("CHILD2").Value)%>">
</td>
</tr>
<tr>
<td width="6%"><font color="#000066" face="Verdana, Arial, Helvetica, sans-serif"
size="2">3.</font></td>
<td width="94%">
<input type="text" name="child3" value="<%=(modify.Fields.Item("CHILD3").Value)%>">
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FFFFFF"><b>Particular
In Property (if any)</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="6%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">1.</font></td>
<td width="94%">
<textarea name="particular1" cols="50"
rows="4"><%=(modify.Fields.Item("PROPERTY1").Value)%></textarea>
</td>
</tr>
<tr>
<td width="6%"><font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-
serif">2.</font></td>
<td width="94%">
<textarea name="particular2" cols="50"
rows="4"><%=(modify.Fields.Item("PROPERTY2").Value)%></textarea>
</td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FFFFFF"><b>Client
Claimed Against The Other Parties</b></font></td>

```

```

</tr>
</table>
<table width="100%" border="0">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#FF0000"><b>Children</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Age</font></td>
<td width="78%">
<input type="text" name="ageclaim" value="<%=(modify.Fields.Item("CHILD_AGE").Value)%>">
</td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Name</font></td>
<td width="78%">
<input type="text" name="nameclaim" value="<%=(modify.Fields.Item("CHILD_NAME").Value)%>">
</td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Date Of Birth</font></td>
<td width="78%">
<input type="text" name="dateclaim" value="<%=(modify.Fields.Item("CHILD_DATE").Value)%>">
</td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Place Of Birth</font></td>
<td width="78%">
<input type="text" name="placeclaim" value="<%=(modify.Fields.Item("CHILD_PLACE").Value)%>">
</td>
</tr>
<tr>
<td width="22%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Sex</font></td>
<td width="78%">
<select name="selectsex">
<option>-SELECT-</option>
<option>Male</option>
<option>Female</option>
<%>
While (NOT modify.EOF)
%>
<option value="<%=(modify.Fields.Item("CHILD_SEX").Value)%>">
<%=(modify.Fields.Item("CHILD_SEX").Value)%></option>
<%>
modify.MoveNext()
Wend
If (modify.CursorType > 0) Then
  modify.MoveFirst
Else
  modify.Requery
End If
%>
</select>
</td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#FF0000"><b>Property</b></font></td>

```

```

</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Type
Of Property 1</font></td>
<td width="72%">
<textarea name="type1" cols="50" rows="4"><%=(modify.Fields.Item("TYPE1").Value)%></textarea>
</td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Location</font></td>
<td width="72%">
<textarea name="location1" cols="50"
rows="4"><%=(modify.Fields.Item("LOCATION1").Value)%></textarea>
</td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Assess</font></td>
<td width="72%"> <font size="2" color="#000066" face="Verdana, Arial, Helvetica, sans-serif">RM
</font>
<input type="text" name="assess1" value="<><%=(modify.Fields.Item("ASSESS1").Value)%>">
</td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Type
Of Property 2</font></td>
<td width="72%">
<textarea name="type2" cols="50" rows="4"><%=(modify.Fields.Item("TYPE2").Value)%></textarea>
</td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Location</font></td>
<td width="72%">
<textarea name="location2" cols="50"
rows="4"><%=(modify.Fields.Item("LOCATION2").Value)%></textarea>
</td>
</tr>
<tr>
<td width="28%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#000066">Assess</font></td>
<td width="72%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">RM
</font>
<input type="text" name="assess2" value="<><%=(modify.Fields.Item("ASSESS2").Value)%>">
</td>
</tr>
</table>
<table width="100%" border="0">
<tr>
<td><font face="Verdana, Arial, Helvetica, sans-serif" size="2"
color="#FF0000"><b>Maintenance</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="33%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">How
much to proposed</font></td>
<td width="67%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-serif">RM
<input type="text" name="proposed" value="<><%=(modify.Fields.Item("PROPOSED").Value)%>">
</font></td>
</tr>
<tr>
<td width="33%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">How
much court decided</font></td>

```

```

<td width="67%"><font color="#000066" size="2" face="Verdana, Arial, Helvetica, sans-serif">RM
<input type="text" name="decided" value="<%=(modify.Fields.Item("DECIDED").Value)%>">
</font></td>
</tr>
</table>
<table width="100%" border="1" bgcolor="#000066">
<tr>
<td><font size="2" color="#FFFFFF" face="Verdana, Arial, Helvetica, sans-serif"><b>Invoice</b></font></td>
</tr>
</table>
<table width="100%" border="1">
<tr>
<td width="41%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Paid</font></td>
<td width="59%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">RM
<input type="text" name="paid" value="<%=(modify.Fields.Item("PAID").Value)%>">
</font></td>
</tr>
<tr>
<td width="41%"><font face="Verdana, Arial, Helvetica, sans-serif" size="2" color="#000066">Any
outstanding amount to be charged</font></td>
<td width="59%"><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#000066">RM
<input type="text" name="baki" value="<%=(modify.Fields.Item("BALANCE").Value)%>">
</font></td>
</tr>
</table>
<p align="right">&nbsp;</p>
<p align="right">&nbsp;</p>
<p>
<input type="hidden" name="MM_update" value="true">
<input type="hidden" name="MM_recordId" value="<%=(modify.Fields.Item("ID").Value)%>">
</p>
</form>
</body>
</html>
<%
modify.Close()
%>

```

UNTUK CARIAN DALAM PENGKALAN DATA

```

<%@LANGUAGE="VBSCRIPT"%>
<!--#include file="Connections/deleteFile.asp" -->
<%
' *** Restrict Access To Page: Grant or deny access to this page
MM_authorizedUsers="admin, Lawyer, Clerk, ADMIN, Admin"
MM_authFailedURL="unauthorizedUtama.asp"
MM_grantAccess=false
If Session("MM_Username") <> "" Then
  If (false Or CStr(Session("MM_UserAuthorization"))="" ) Or
    (InStr(1,MM_authorizedUsers,Session("MM_UserAuthorization"))>=1) Then
      MM_grantAccess = true
    End If
End If
If Not MM_grantAccess Then
  MM_qsChar = "?"
  If (InStr(1,MM_authFailedURL,"?") >= 1) Then MM_qsChar = "&"
  MM_referrer = Request.ServerVariables("URL")
  if (Len(Request.QueryString()) > 0) Then MM_referrer = MM_referrer & "?" & Request.QueryString()
  MM_authFailedURL = MM_authFailedURL & MM_qsChar & "accessdenied=" &
  Server.URLEncode(MM_referrer)
  Response.Redirect(MM_authFailedURL)
End If
%>
<%
set deleteFile = Server.CreateObject("ADODB.Recordset")

```

```

deleteFile.ActiveConnection = MM_deleteFile_STRING
deleteFile.Source = "SELECT * FROM insertData"
deleteFile.CursorType = 0
deleteFile.CursorLocation = 2
deleteFile.LockType = 3
deleteFile.Open()
deleteFile_numRows = 0
%>
<html>
<head>
<title>search_record</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000" background="images/background.gif">
<form name="form1" method="get" action="displayFile.asp">
<table width="100%" border="0" bgcolor="#000066">
<tr>
<td>
<div align="center"><font size="4" face="Verdana, Arial, Helvetica, sans-serif"><b><font
color="#FFFFFF" size="5">SEARCH
RECORD</font></b></font></div>
</td>
</tr>
</table>
<p>&nbsp;</p>
<table width="60%" border="0" align="center" bgcolor="#003399">
<tr>
<td width="3%">&nbsp;</td>
<td width="49%">&nbsp;</td>
<td width="46%">&nbsp;</td>
<td width="2%">&nbsp;</td>
</tr>
<tr>
<td width="3%">&nbsp;</td>
<td width="49%"><b><font size="2" face="Verdana, Arial, Helvetica, sans-serif" color="#FFFFFF">Search
By File Reference :</font></b></td>
<td width="46%">
<select name="select_File">
<%
While (NOT deleteFile.EOF)
%>
<option
value=<%=(deleteFile.Fields.Item("ID").Value)%>><%= (deleteFile.Fields.Item("FILE_REFERENCE").Value
)%></option>
<%
deleteFile.MoveNext()
Wend
If (deleteFile.CursorType > 0) Then
  deleteFile.MoveFirst
Else
  deleteFile.Requery
End If
%>
</select>
</td>
<td width="2%">&nbsp;</td>
</tr>
<tr>
<td width="3%">&nbsp;</td>
<td width="49%">&nbsp;</td>
<td width="46%">&nbsp;</td>
<td width="2%">&nbsp;</td>
</tr>
</table>
<div align="center">
<p>&nbsp;</p>
<p>

```

```

<input type="submit" name="go" value="GO!">
</p>
</div>
</form>
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
</body>
</html>
<%
deleteFile.Close()
%>

```

PENGKODAN JAVASCRIPT

//Static Slide Menu 6.5 © MaXimuS 2000-2001, All Rights Reserved.

//Site: http://www.absolutegb.com/maximus

//E-mail: maximus@nsimail.com

//Script featured on Dynamic Drive (http://www.dynamicdrive.com)

```

NS6 = (document.getElementById&&!document.all)
IE = (document.all)
NS = (navigator.appName=="Netscape" && navigator.appVersion.charAt(0)=="4")

tempBar="";barBuilt=0;ssmItems=new Array()

moving=setTimeout('null',1)
function moveOut() {
if ((NS6||NS)&&parseInt(ssm.left)<0 || IE && ssm.pixelLeft<0) {
clearTimeout(moving);moving = setTimeout('moveOut()', slideSpeed);slideMenu(10)}
else {clearTimeout(moving);moving=setTimeout('null',1)}}
function moveBack() {clearTimeout(moving);moving = setTimeout('moveBack1()', waitTime)}
function moveBack1() {
if ((NS6||NS) && parseInt(ssm.left)>(-menuWidth) || IE && ssm.pixelLeft>(-menuWidth)) {
clearTimeout(moving);moving = setTimeout('moveBack1()', slideSpeed);slideMenu(-10)}
else {clearTimeout(moving);moving=setTimeout('null',1)}}
function slideMenu(num){
if (IE) {ssm.pixelLeft += num;}
if (NS||NS6) {ssm.left = parseInt(ssm.left)+num;}
if (NS) {bssm.clip.right+=num;bssm2.clip.right+=num;}}

function makeStatic() {
if (NS||NS6) {winY = window.pageYOffset;}
if (IE) {winY = document.body.scrollTop;}
if (NS6||IE||NS) {
if (winY!=lastY&&winY>YOffset-staticYOffset) {
smooth = .2 * (winY - lastY - YOffset + staticYOffset);}
else if (YOffset-staticYOffset+lastY>YOffset-staticYOffset) {
smooth = .2 * (winY - lastY - (YOffset-(YOffset-winY));)}
else {smooth=0}
if(smooth > 0) smooth = Math.ceil(smooth);
else smooth = Math.floor(smooth);
if (IE) bssm.pixelTop+=smooth;
if (NS6||NS) bssm.top=parseInt(bssm.top)+smooth
lastY = lastY+smooth;
setTimeout('makeStatic()', 1)}}}

function buildBar() {
if(barText.indexOf('<IMG>')>-1) {tempBar=barText}
else{for (b=0;b<barText.length;b++) {tempBar+=barText.charAt(b)+"<BR>"}}
document.write('<td align="center" rowspan="100" width="'+barWidth+'" bgcolor="'+barBGColor+'"
valign="'+barVAlign+'><p align="center"><font face="'+barFontFamily+'"
Size="'+barFontSize+'"
COLOR="'+barFontColor+'<B>' +tempBar+'</B></font></p></TD>')

function initSlide() {
if (NS6){ssm=document.getElementById("thessm").style;bssm=document.getElementById("basessm").style;

```

```

bssm.clip="rect(0 "+document.getElementById("thessm").offsetWidth+
"+document.getElementById("thessm").offsetHeight+" 0);ssm.visibility="visible";
else if (IE) {ssm=document.all("thessm").style;bssm=document.all("basessm").style
bssm.clip="rect(0 "+thessm.offsetWidth+" "+thessm.offsetHeight+" 0);bssm.visibility = "visible";
else if (NS) {bssm=document.layers["basessm1"];
bssm2=bssm.document.layers["basessm2"];ssm=bssm2.document.layers["thessm"];
bssm2.clip.left=0;ssm.visibility = "show";
if (menuIsStatic=="yes") makeStatic();}

function buildMenu() {
if (IE||NS6) {document.write('<DIV ID="basessm" style="visibility:hidden;Position : Absolute ;Left :
'+XOffset+' ;Top : '+YOffset+' ;Z-Index : 20;width:'+(menuWidth+barWidth+10)+"><DIV ID="thessm"
style="Position : Absolute ;Left : '+(-menuWidth)+" ;Top : 0 ;Z-Index : 20;" onmouseover="moveOut()"
onmouseout="moveBack()">');
if (NS) {document.write('<LAYER name="basessm1" top="'+YOffset+" LEFT='"+XOffset+
visibility="show"><ILAYER name="basessm2"><LAYER visibility="hide" name="thessm"
bgcolor="'+menuBGColor+" left="'+(-menuWidth)+" onmouseover="moveOut()
onmouseout="moveBack()">');
if (NS6){document.write('<table border="0" cellpadding="0" cellspacing="0"
width="'+(menuWidth+barWidth+2)+"' bgcolor="'+menuBGColor+"><TR><TD>');
document.write('<table border="0" cellpadding="0" cellspacing="1" width="'+(menuWidth+barWidth+2)+"'
bgcolor="'+menuBGColor+">');
for(i=0;i<ssmItems.length;i++) {
if(!ssmItems[i][3]) {ssmItems[i][3]=menuCols;ssmItems[i][5]=menuWidth-1}
else if(ssmItems[i][3]!=menuCols) ssmItems[i][5]=Math.round(menuWidth*(ssmItems[i][3]/menuCols)-1);
if(ssmItems[i-1] && ssmItems[i-1][4]!='no') {document.write('<TR>');
if(!ssmItems[i][1]){
document.write('<td bgcolor="'+hdrBGColor+" HEIGHT="'+hdrHeight+'" ALIGN="'+hdrAlign+
VALIGN="'+hdrVAlign+'" WIDTH="'+ssmItems[i][5]+'' COLSPAN="'+ssmItems[i][3]+''>&nbsp;<font
face="'+hdrFontFamily+'" Size="'+hdrFontSize+'"
COLOR="'+hdrFontColor+'><b>' +ssmItems[i][0]+'</b></font></td>');
else {if(!ssmItems[i][2]) ssmItems[i][2]=linkTarget;
document.write('<TD BGCOLOR="'+linkBGColor+" onmouseover="bgColor='\\"+linkOverBGColor+'\"
onmouseout="bgColor='\\"+linkBGColor+'\\" WIDTH="'+ssmItems[i][5]+'' COLSPAN="'+ssmItems[i][3]+''><ILAYER><LAYER onmouseover="bgColor='\\"+linkOverBGColor+'\"
onmouseout="bgColor='\\"+linkBGColor+'\\" WIDTH="100%" ALIGN="'+linkAlign+'><DIV
ALIGN="'+linkAlign+'><FONT face="'+linkFontFamily+'" Size="'+linkFontSize+'>&nbsp;<A
HREF="'+ssmItems[i][1]+'" target="'+ssmItems[i][2]+'"'
CLASS="'+ssmItems[i][4]>' +ssmItems[i][0]+'

```

<!--

/*

Configure menu styles below

NOTE: To edit the link colors, go to the STYLE tags and edit the ssm2Items colors

*/

YOffset=150; // no quotes!!

XOffset=0;

staticYOffset=30; // no quotes!!

slideSpeed=20 // no quotes!!

waitTime=100; // no quotes!! this sets the time the menu stays out for after the mouse goes off it.

menuBGColor="black";

menuIsStatic="yes"; //this sets whether menu should stay static on the screen

menuWidth=130; // Must be a multiple of 10! no quotes!!

menuCols=2;

hdrFontFamily="verdana";

hdrFontSize="2";

hdrFontColor="white";

hdrBGColor="#000066";

```
hdrAlign="left";
hdrVAlign="center";
hdrHeight="15";
linkFontFamily="Verdana";
linkFontSize="2";
linkBGColor="white";
linkOverBGColor="#FFFF99";
linkTarget="_blank";
linkAlign="Left";
barBGCOLOR="#000066";
barFontFamily="Verdana";
barFontSize="2";
barFontColor="white";
barVAlign="center";
barWidth=20; // no quotes!
barText="SIDE MENU"; // <IMG> tag supported. Put exact html for an image to show.

///////////

// ssmItems[...]=[name, link, target, colspan, endrow?] - leave 'link' and 'target' blank to make a header
ssmItems[0]=["Document"] //create header
ssmItems[1]=["System Overview", "http://localhost/norzita/overview.asp", "mainFrame"]
ssmItems[2]=["Legal Form", "http://localhost/norzita/link_form.asp", "mainFrame"]
ssmItems[3]=["Article", "http://localhost/norzita/article_menu.asp", "mainFrame"]
//ssmItems[4]=["FAQ", "http://IS.sapura.com.my", "", 1, "no"] //create two column row
//ssmItems[5]=["Email", "http://IS.sapura.com.my", "",1]
//Renumber the item below if you unremark the "FAQ" and "Email"
ssmItems[4]=["Client Particular"]//create header
ssmItems[5]=["Insert New Record", "http://localhost/norzita/form_insertdata.asp", "mainFrame"]
//ssmItems[4]=["FAQ", "http://IS.sapura.com.my", "", 1, "no"] //create two column row
//ssmItems[5]=["Email", "http://IS.sapura.com.my", "",1]
//Renumber the item below if you unremark the "FAQ" and "Email"
ssmItems[6]=["Search", "", "" ] //create header
ssmItems[7]=["By File Reference", "http://localhost/norzita/search_record_byFile.asp", "mainFrame"]
ssmItems[8]=["By Client Name", "http://localhost/norzita/search_record_byName.asp", "mainFrame"]

buildMenu();

//-->
```

RUJUKAN

University of Malaya

RUJUKAN

a. Laporan tesis

1. Nor Fadhlina Wati Said, (2000), “**Sistem Maklumat Pesakit**”, Fakulti Sains Komputer Dan Teknologi Maklumat, Universiti Malaya, Kuala Lumpur.
2. Mat Zam Jarkoni, (2000), “**Test Bank**”, Fakulti Sains Komputer Dan Teknologi Maklumat, Universiti Malaya, Kuala Lumpur.
3. Nor Amani Abdul Rashid, (2000), “**Pakej Sistem Pendaftaran Kursus Atas-Talian**”, Fakulti Teknologi Dan Sains Maklumat Universiti Kebangsaan Malaysia, Bangi, Selangor.

b. Halaman Web

1. <http://www.courtreporter.law.com/>
2. <http://www.findlaw.com/>
3. <http://www.iplcca.com/>
4. <http://www.lawnet.com/>
5. <http://www.asp1001.com>
6. <http://www.askasp.com>
7. <http://www.asptoday.com>
8. <http://www.e-SAKINAH.com>

c. **Buku Penerbitan**

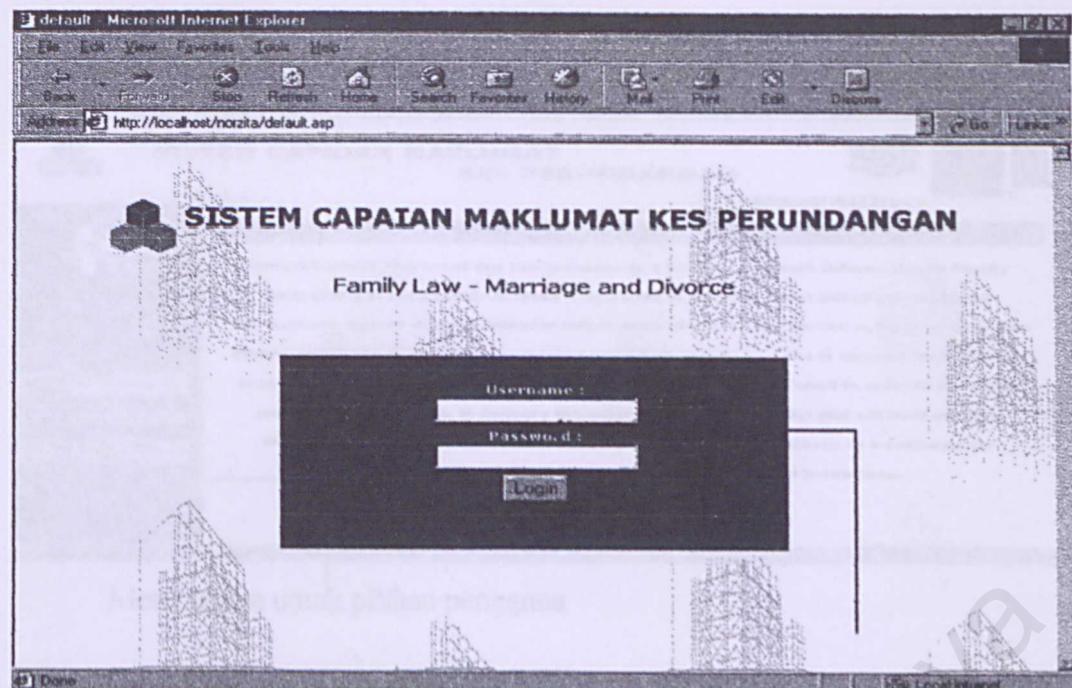
1. Ian Sommerville, (1996), “**Software Engineering**”, Fifth Edition, Addison Wiley, United State Of America
2. Shari Lawrence Pfleeger, (2001), “**Software Engineering-Theory and Practice**”, Second Edition, Prentice Hall International, United State Of America.
3. Philip L.Weaver, Nick Lambrou and Matthew Walkley. 1998. **Practical SSADM Version 4+, A Complete Tutorial Guide 2nd Edition.** London: Financial Times Pitman Publishing.
4. Unit Perhubungan Hal Ehwal Pelajar , UKM.1998. **Panduan Pelajar .** Selangor:Unit Perhubungan Pelajar UKM.
5. David M.Kroenke. 1995 . **Database Processing 5th Edition.**United States: Prentice Hall.

MANUAL PENGGUNA

University of Malaya

MODUL PENGGUNA “CLERK” DAN STAFF BIASA

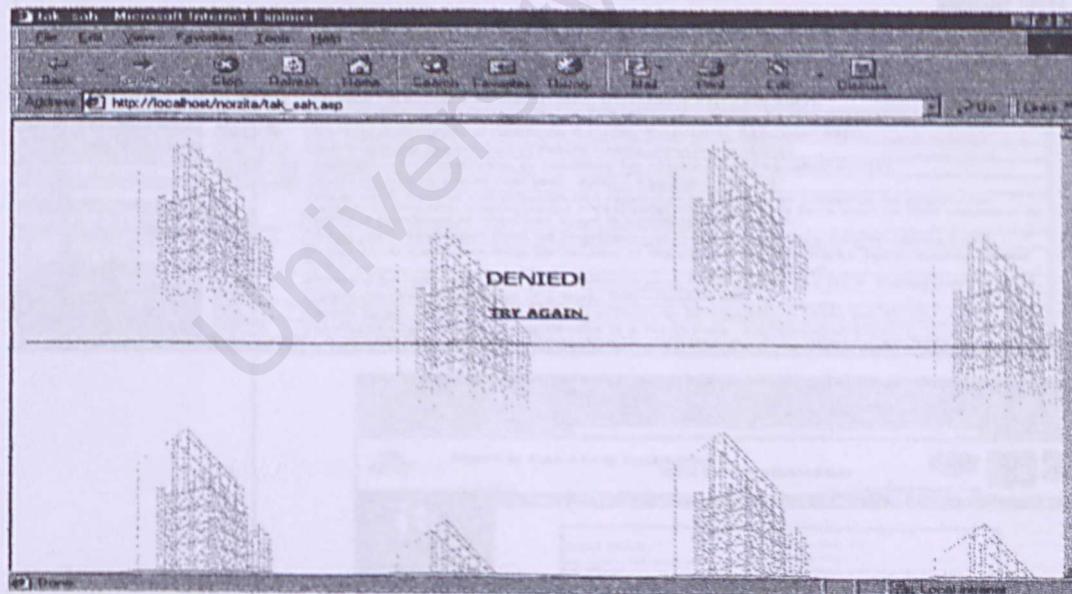
Universiti Malaysia Sarawak
Jalan Gurney, 88400 Kuching, Sarawak, Malaysia
E-mail: clerk@ums.edu.my | www.ums.edu.my/clerk



Tekan setelah selesai

memasukkan username dan password

Adalah antaramuka pertama yang akan dipaparkan setiap kali pengguna menggunakan sistem.



Dipaparkan apabila proses "LOGIN" tidak berjaya disebabkan kesalahan "USERNAME" atau "PASSWORD"

| |
|---|
|
SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

OVERVIEW <p>Sistem Capaihan Maklumat Kes Perundangan is a Virtual Law Client Information In Family Law. It is a system which includes a database of updated client informations. These informations include all the particular information about the maintenance, child custody and other parties information. It is comprehensive and easy to use and is catered for practising lawyers, legal advisors and legal firm staff who demand a daily need to refer to the client particular information. It delivers the most convenient, low cost and efficient solution for users through direct networking and for quick and easy access to a continuously expanding database without time and geographical limitations.</p> |
|---|

Menu utama untuk pilihan pengguna

Antaramuka ke dua setelah proses “LOGIN” berjaya. Antaramuka mengandungi menu utama yang lengkap bagi keseluruhan sistem. Hanya perlu klik pada “SIDE MENU” untuk pergi ke laman seterusnya.

Untitled Document - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Options

<http://localhost/moraktaAutama.asp>

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

03/06/2003 00:25:22 Local

LEGAL FORMS

| | |
|----------|---|
| FORM 1 | Request for grant of Patent; Regulation 7(1). |
| FORM 1A | Request for substantive examination; Regulations 27(1) and 48(3). |
| FORM 5A | Request for modified substantive examination; Regulations 27A(1) and 45(3). |
| FORM 1B | Application for grant of certificate for utility innovation; Regulation 45(1). |
| FORM 1B | Application for registration of Patent Agents; Regulation 45C(1). |
| FORM 10A | Application for registration as candidate for examination; Regulation 45D(2). |
| FORM 10B | Application to re-site examination; Patent; Regulation 45D(3). |
| TM 1 | Trade mark form of authorization and request to enter, alter or substitute an address for service in Malaysia; subregulation 10(1), 10(2) and 11(2); this form must be filed whenever an agent is appointed or when one agent is substituted for another. |
| TM 2 | Trade mark application form for registration of Trade Marks Agent; Subregulations 12(2). |
| TM 3 | Trade mark application form for renewal of registration of Trade Marks Agent; Subregulations 12(7). |
| TM 4 | Request for registrar's Preliminary Advice as to Registrability of a Mark; Subregulations 17(1). |
| TM 5 | Application For Registration of a Mark; Subregulation 18(1). |
| TM 6 | Notice Is the registrar of Attendance at hearing; Subregulations 50(4) and 51(1). |
| TM 7 | Application for renewal of Registration of a Trade Mark; Subregulation 57(1). |

Halaman mengandungi borang-borang rasmi yang digunakan di dalam prosedur mahkamah. Hanya klik pada perkataan hyperlink untuk membuka halaman borang.

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://localhost/norzilla/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

BACK TO MENU

FAMILY LAW

Document System Overview Legal Form Article Client Particular Insert New Record Search By File Reference By Client Name

FAMILY LAW FOR NON-MUSLIMS IN MALAYSIA

B Lee Swee Seng LLB, LLM, MBA Advocate & Solicitor High Court, Malaysia Certified Mediator Trade Marks Agent Industrial Designs Agent

1.0 Introduction

1.1. The family remains the basic unit of society. However, never before has the institution of the family come under such severe stress and strain. Laws have to be put in place for the proper ordering of society. There are many statutes governing non-Muslim family life in Malaysia such as the Married Women and Children (Maintenance) Act 1950, the Adoption Act 1952, the Distribution Act 1958, the Wills Act 1959, the Legitimacy Act 1961, the Guardianship of Infants Act 1971, the Law Reform (Marriage and Divorce) Act 1976 and the Domestic Violence Act 1994, to name but a few. Time does not permit us to cover the main provisions of every piece of legislation. As for the Wills Act 1959, a treatment of it is given in Appendix A and a sample Will in Appendix B. The summarised table of distribution of one's estate upon death without a Will is given in Appendix C. Much of the focus of this paper on Family Law in Malaysia will be on the problem areas in the ordering of the family life under the Law Reform (Maintenance and Divorce) Act 1976.

Untuk antaramuka artikel yang mengandungi tulisan artikel oleh tokoh-tokoh atau peguam-pegawai yang bertauliah di seluruh Malaysia. Ini digunakan sebagai bahan bacaan kepada para pengguna sistem.

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://localhost/norzilla/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

03/06/2003 00:29:47 Local

FAMILY LAW

Document System Overview Legal Form Article Client Particular Insert New Record Search By File Reference By Client Name

YOU ARE UNAUTHORIZED TO PERFORM THIS PAGE!

tekan untuk kembali ke halaman
“Overview”

Pengguna biasa atau staf lain selain peguam dan kerani tidak boleh mengakses halaman “INSERT NEW RECORD”, “BY FILE REFERENCE” dan “BY CLIENT NAME” sebagai langkah keselamatan sistem.

Untitled Document - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://localhost/norzilla/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

02/06/2003 00:42:02 Local

CLIENT PARTICULAR

File Reference
Status Of Case
Client Particular

| | |
|-----------------------|----------------------|
| Name | <input type="text"/> |
| Identification Number | <input type="text"/> |
| Address | <input type="text"/> |

FAMILY LAW

Document System Overview Legal Form Article Client Particular Insert New Record Search By File Reference By Client Name

Paparan “Insert New Record” yang boleh diakses oleh Administrator, Lawyer dan Clerk sahaja. Isi borang yang terdapat di dalam halaman, dan data akan dihantar untuk di simpan di dalam pengkalan data dan akan memaparkan mesej pemberitahuan data telah disimpan seperti di bawah

Untitled Document - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://localhost/norzilla/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

02/06/2003 00:53:59 Local

DATA HAS BEEN STORED

RETURN TO NEW FORM

Klik untuk memasukkan lagi data maklumat

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History Mail Print Edit Options

Address http://localhost/norzita/utama.asp 02/06/2003 00:57:43 Local

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

SEARCH RECORD

Search By File Reference : **12345**

12345
45454

GOK

Tekan setelah selesai memilih

List Menu

Merupakan halaman bagi carian rekod data di dalam pengkalan data dengan menggunakan kata kunci nombor rujukan. Pengguna hanya perlu pilih daripada "LIST MENU" yang terdapat pada halaman.

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History Mail Print Edit Options

Address http://localhost/norzita/utama.asp 02/06/2003 01:03:54 Local

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

UPDATE MAIN DELETE

CLIENT PARTICULAR

| | |
|-------------------------------------|-------------|
| File Reference | 12345 |
| Status Of Case | klikliklikh |
| Client Particular | |
| Name | norzita |
| Identification Number | 676766 |
| Address | gh00000000 |
| The Other Party's Particular | |
| Name | jhgjhgjhjh |
| Identification Number | 2212222 |
| Address | hgjhjhjhjh |
| Solicitor Name and Address | |
| hgjgjhgkkjhjhjhjh | |

teks hijau tanda katakunci

Paparan rekod yang diambil daripada pengkalan data. Teks yang diberikan anak panah menunjukkan carian dilakukan berdasarkan kata kunci tersebut.

KES PERUNDANGAN

02/06/2003 01:03:5

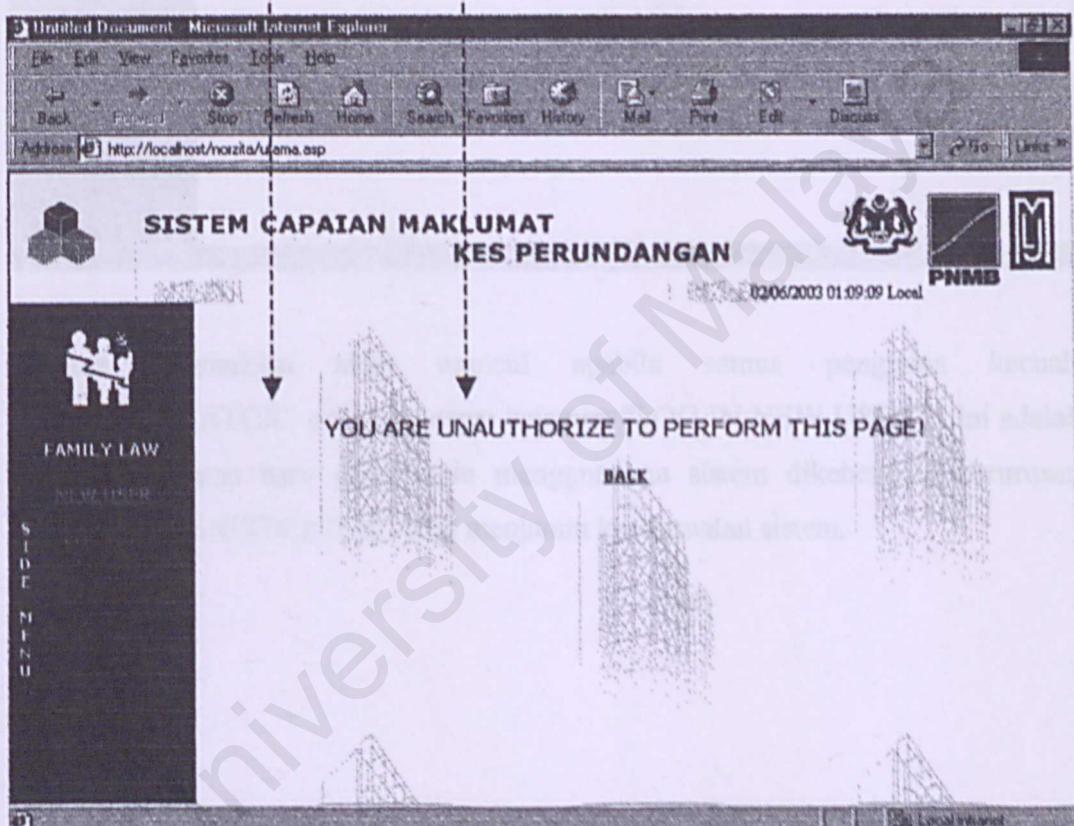
UPDATE MAIN DELETE

CLIENT PARTICULAR

12345

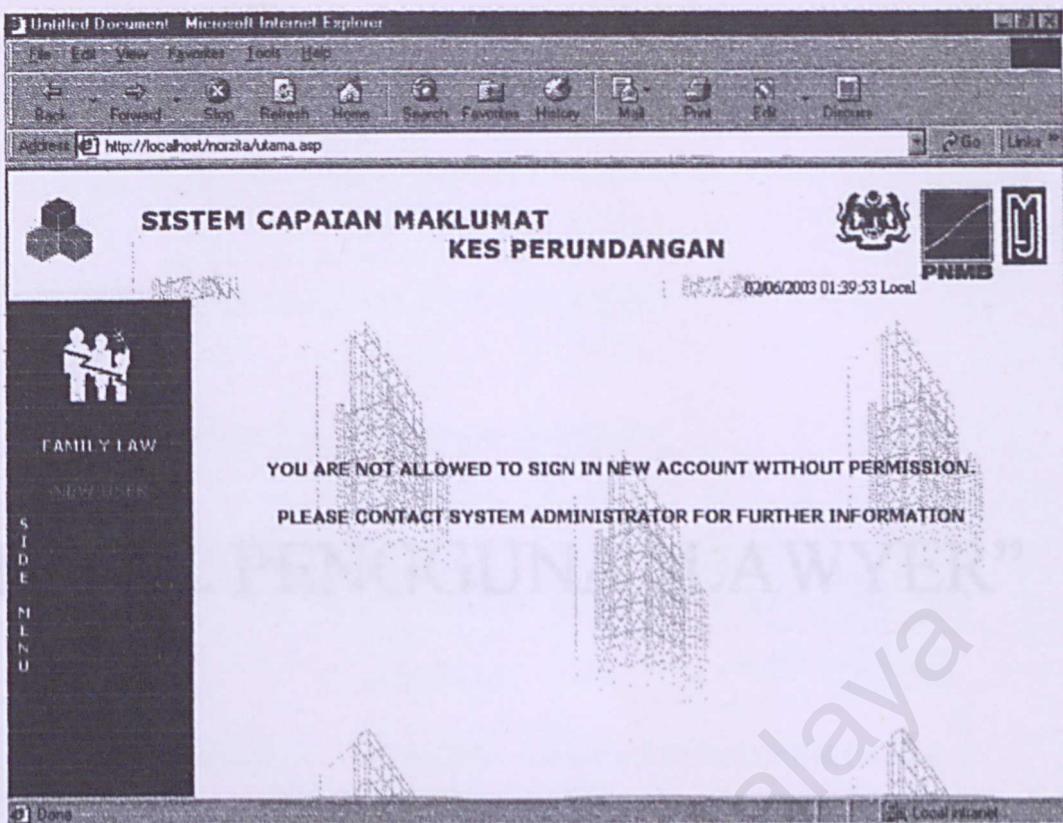
klhklhkhjh

norrita



Sistem akan memaparkan mesej di atas setiap kali pengguna biasa dan clerk menekan butang “UPDATE” dan “DELETE”. Ini dilakukan sebagai keselamatan supaya maklumat di dalam pengkalan data tidak diubah atau dihapuskan tanpa pengetahuan peguam atau Administrator.

Maka pengguna “CLERK” dan lain staf hanya boleh mencapai sistem setakat ini sahaja.



Paparan sedemikian akan muncul apabila semua pengguna kecuali “ADMINISTRATOR” cuba mencapai halaman “LOG IN NEW USER”. Ini adalah kerana, pengguna baru yang ingin menggunakan sistem dikehendaki berurusan dengan “ADMINISTRATOR” bagi menjamin keselamatan sistem.

Pengguna "LAWYER" menggunakan fungsi dan perintah sistem yang tidak dapat dicapai oleh pengguna "CLERK" sebelum di buat.

MODUL PENGGUNA "LAWYER"

Pengguna halaman "UPDATE" untuk mengelola data "LAWYER". Setelah data diambil, menu "UPDATE RECORD" pada baris untuk memasukan perubahan data. Halaman di bawah akan muncul apabila data yang dimasukkan.

Pengguna "LAWYER" mempunyai tambahan capaian sistem yang tidak dapat dicapai oleh pengguna "CLERK" seperti di bawah.

KES PERUNDANGAN

02/06/2003 01:03:5

UPDATE MAIN DELETE

CLIENT PARTICULAR

12345
kikhkhjh

norzita

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address: 02 http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

MAIN BACK

02/06/2003 01:27:03 Local

CLIENT PARTICULAR

File Reference: 12345
Status Of Case: kikhkhjh

Client Particular

| | |
|------------------------|---------|
| Name: | norzita |
| Identification Number: | 676788 |
| | aaaaaaa |

Tekan setelah selesai kemaskini

Paparan halaman "UPDATE" yang boleh dicapai oleh "LAWYER". Setelah data diubah, butang "UPDATE RECORD" perlu ditekan untuk mngemaskini pengkalan data. Halaman di bawah akan muncul apabila data berjaya dikemaskini.

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address: 02 http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

02/06/2003 01:30:44 Local

OK!

RETURN

FAMILY LAW

REVIEW CASE

KES PERUNDANGAN

02/06/2003 01:03:5.

UPDATE MAIN DELETE

CLIENT PARTICULAR

12345
klhkhkjh

Inorzita

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Options

Address http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

PEREX

03/06/2003 01:33:28 Local

BACK MAIN

CLIENT PARTICULAR

DELETE RECORD

File Reference: 12
Status Of Case: klhkhkjh

Client Particular

| | |
|-----------------------|------------|
| Name | Inorzita |
| Identification Number | 676786 |
| Address | ghgggggggg |

The Other Party's Particular

| | |
|----------------------------|-------------------|
| Name | jhgjhgjhjhjh |
| Identification Number | 2212222 |
| Address | hgjhjhjhjh |
| Solicitor Name and Address | hjgjgjhkkjhjhjhjh |

Untuk menghapuskan rekod, hanya kilik butang “DELETE RECORD” dan rekod akan dihapuskan dari pengkalan data. Setelah proses berjaya, halaman di bawah akan muncul.

Untitled Document Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Options

Address http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

PEREX

03/06/2003 01:30:44 Local

OK!

RETURN

"ADMINISTRATOR" mempunyai ciri-ciri seperti yang diatas. Beliau mampu membuat pengemaskinian dan perbaharuan rekod dan percasian dan. Agak macam jadi pengguna kepada "LOG IN NEW USER".

SISTEM CAIRAN PENGETAHUAN
PERSEKUTUAN

MODUL PENTADBIR

Misi untuk seleksi

Pengguna "LOG IN NEW USER" yang dapat dibuat oleh "ADMINISTRATOR". Detalah seorang pengguna "LOG IN NEW USER" diberikan untuk menyaksikan maklumat pengguna baru tiba ke dalam penggunaan.

“ADMINISTRATOR” mendapat capaian sepenuhnya ke atas sistem. Boleh melakukan pengemaskinian dan penghapusan rekod dari pengkalan data. Juga menjadi pengurus kepada “LOG IN NEW USER”.

The screenshot shows a Microsoft Internet Explorer window with the title bar "Untitled Document - Microsoft Internet Explorer". The address bar contains "http://localhost/norzita/utama.asp". The main content area displays a web page titled "SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN". On the right side of the header are the Malaysian coat of arms and the PNMB logo. Below the title is a timestamp "02/06/2003 01:44:00 Local". The central part of the page is a "NEW USER SIGN IN" form with fields for "Username", "Password", and "Position" (with a dropdown menu showing "-SELECT-"). A "Submit" button is at the bottom. To the left of the form is a sidebar with icons for "FAMILY LAW", "NEW USER", and "S T D E M E N U". At the bottom of the page is a "RETURN TO LOGIN" link.

klik setelah selesai

Paparan “LOG IN NEW USER” yang dapat dicapai oleh “ADMINISTRATOR”. Setelah selesai, butang “SUBMIT” ditekan untuk penyimpanan maklumat pengguna baru sistem ke dalam pengkalan data.

Untitled Document - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address: http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

02/06/2003 01:46:52 Local

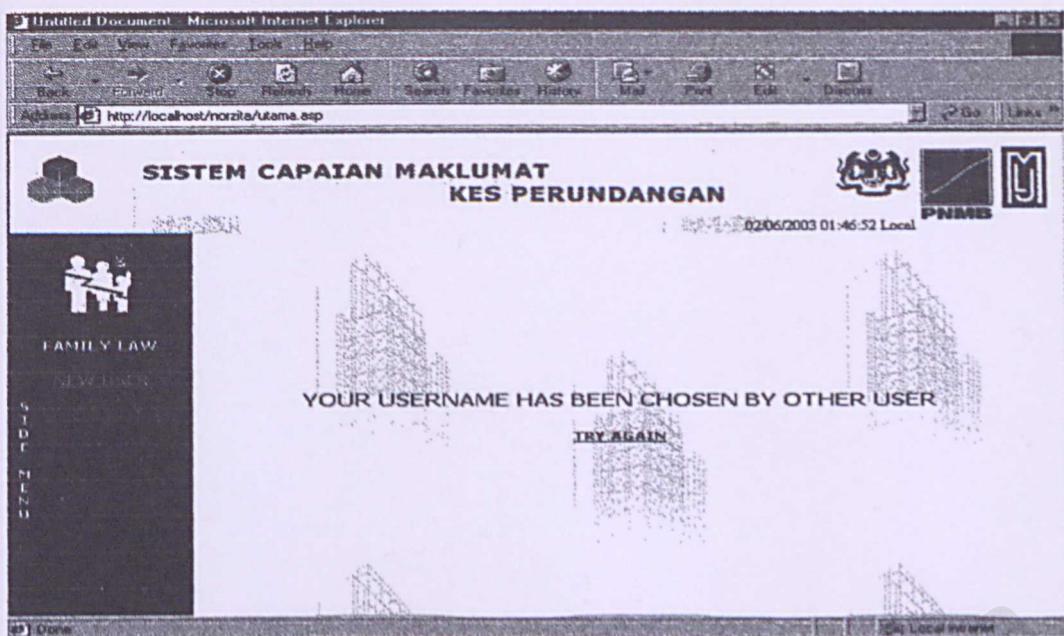
Family Law

YOUR USERNAME HAS BEEN CHOSEN BY OTHER USER

TRY AGAIN

PNMB

SIDE MENU



“USERNAME” pengguna sistem seharusnya unik supaya tidak berlaku ‘konflik’ di dalam capaian semula maklumat di pengkalan data. Maka, skrin di atas akan muncul sekiranya pengguna baru sistem memasukkan “USERNAME” yang telah digunakan oleh pengguna lain. Skrin akan sentiasa muncul sehingga pengguna memasukkan “USERNAME” yang tidak sama dengan pengguna lain.

Untitled Document - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address: http://localhost/norzita/utama.asp

SISTEM CAPAIAN MAKLUMAT KES PERUNDANGAN

03/06/2003 01:50:25 Local

Family Law

DATA HAS BEEN STORED.

OK

PNMB

SIDE MENU



Paparan sekiranya pengguna berjaya memasukkan “USERNAME” yang unik.