|  |  |  |
| --- | --- | --- |
| **BAHAGIAN PENDIDIKAN DAN LATIHAN TEKNIK VOKASIONAL**  **KEMENTERIAN PENDIDIKAN MALAYSIA**  **ARAS 5 & 6, BLOK E14, KOMPLEKS E,**  **PUSAT PENTADBIRAN KERAJAAN PERSEKUTUAN**  **KERTAS PENERANGAN**  ***(INFORMATION SHEET)***  C:\Users\norsu\Desktop\jata negara.png | | |
| **KOD DAN NAMA PROGRAM NOSS** | IT-010-3:2016 PEMBANGUNAN APLIKASI | |
| **TAHAP NOSS** | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | **CU05/WA3 - DEBUG APPLICATION CODE** | |
| **NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD2033 FUNDAMENTAL OF PROGRAMMING | |
| **NO DAN TAJUK STANDARD KANDUNGAN KSKV** | K3 WRITE MODULE CODE | |
| **NO. KOD NOSS** | IT-010-3:2016-C05 / P(25/27) | Muka Surat : 1 Drp : 19 |
| **NO. KOD KSKV** | KPD2033/P(11/13) |

**TAJUK/***TITLE***: PENGUJIAN UNIT**

**TUJUAN/***PURPOSE***:**

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

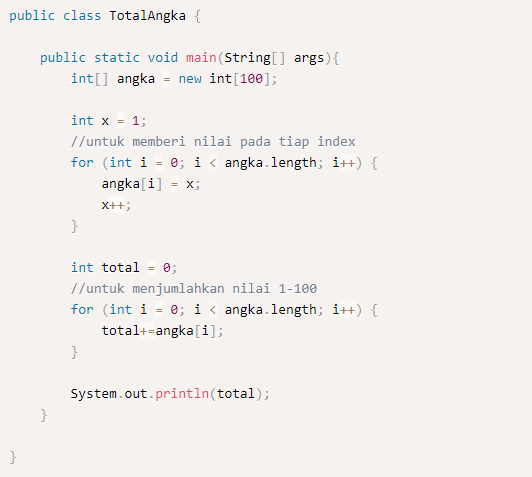
* Struktur Data dan Modular
* Pengujian Unit
* Prosidur Nyahpepijat

**PENERANGAN:**

**PENGERTIAN ARRAY**

**Array adalah suatu tipe data terstruktur yang berupa sejumlah data sejenis (bertipe data sama) yang jumlahnya tetap dan diberi suatu nama tertentu.**Setiap komponen atau elemen array dapat diakses dan dibedakan melalui indeks yang spesifik dimana jumlahnya sebanyak ukuran array tersebut dikurangi satu (kerana perhitungan indeks dimulai darinol).Jumlah elemen terbatas, dan semua elemenmempunyai tipe yang sama.

untuk menghitung indeks array misalnya array dengan indeks 5, maka memori yang tersedia dihitung dari indeks ke 0 .



Rajah1: contoh program array :

Disana kita mendeklarasikan 3 pembolehubah:

* Yang pertama, sebuah pembolehubah l array bernama angka dengan jenis integer.
* Yang kedua pembolehubah x jenis data integer dengan nilai awal = 1.
* Yang ketiga, adalah pembolehubah total untuk menyimpan jumlah nilai.

Kemudian kita menggunakan fungsi perulangan atau looping.

* Looping pertama untuk memberikan nilai untuk tiap index array dari 1-100.
* Looping kedua kita gunakan untuk menjumlahkan tiap nilai kedalam variabel total.

Lalu, diakhir program kita akan menampilkannya hasil dari penjumlahannya.

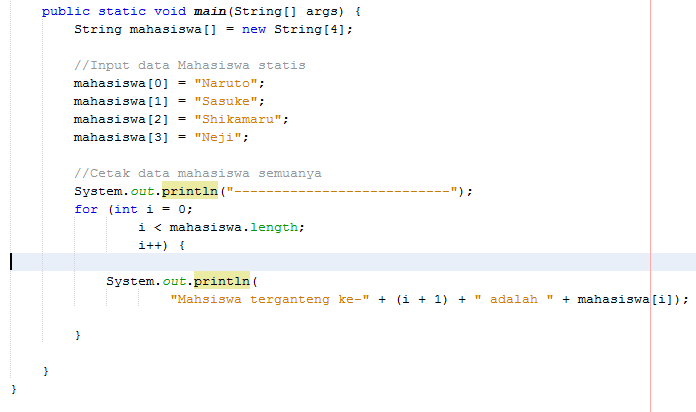
Array juga ada 2 jenis,iaitu:

\***Array 1 dimensi**:

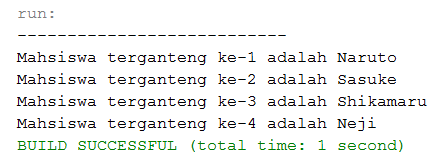
**Array 1 dimensi**,kerana memerlukan hanya 1 baris saja

Sebagai langkah efisiensi penggunan memori komputer,sebab data elemen array di lokasikan pada suatu deretan sel memori tertentu. Hal ini jauh lebih efisien dibandingkan dengan jika kita mendefinisikan masing -masing data pada suatu pembolehubah tersendiri yang pastinya akan banyak mrnyimpan memori.

**contoh array 1 dimensi**:

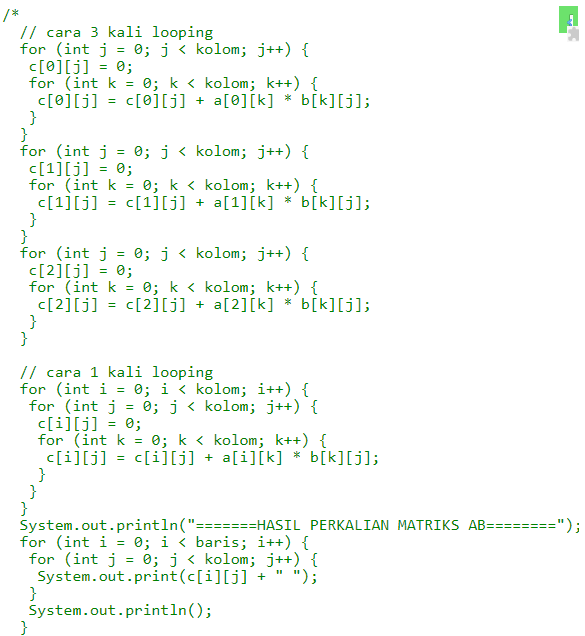


Rajah 2: **contoh array 1 dimensi**:



Rajah 3: **Hasil output** :

Contoh **array 2 dimensi:**



Rajah 3: **contoh array 2 dimensi**:

**Jadi manfaat kita menggunakan array**:

1. Mempermudah pengelolaan atau pengembangan program mengingat setiap subrutin memiliki kode yang relatif sedikit jika dibandingkan dengan kode program secara keseluruhan yang disusun tanpa melibatkan subrutin.
2. Dapat digunakan untuk mengurangi jumlah kode akibat sejumlah kode yang sama digunakan beberapa kali dalam program.
3. Menjadikan program lebih bersifat modular sehingga akan mudah difahami dan dapat digunakan kembali, baik untuk program itu sendiri maupun untuk program lain yang memiliki proses yang sama.

**PENGUJIAN UNIT**

**UJIAN UNIT adalah tahap pengujian perisian di mana unit / komponen individu dari suatu perisian diuji**. Tujuannya adalah untuk mengesahkan bahawa setiap unit perisian berfungsi seperti yang direka. Satu unit adalah bahagian yang paling kecil yang dapat diuji dari mana-mana perisian. Ia biasanya mempunyai satu atau beberapa input dan biasanya satu output.

Pegujian dilakukan terhadap unit perisian/ laman web terkecil yang dinamakan modul. Modul terdiri daripada satu fungsi atau prosedur atau satu-satu laman web. Setiap modul perlu diuji dari beberapa aspek seperti pengendali ralat, antaramuka, navigasi dan lain-lain

Pengendali ralat

antaramuka

**MODUL**

Laluan logic / navigasi

Struktur data

Nilai sempadan & kelas data

**Rajah 4: Jenis-jenis pengujian**

**PENGUJIAN KEBOLEHGUNAAN (*USABILITY TESTING*)**

Kebolehgunaan terhadap aplikasi yang dibangunkan hanya boleh ditentukan apabila pengujian kebolehgunaan dilaksanakan. Pembangun boleh melihat pengguna berinteraksi dengan aplikasi dan pengguna akan menjawab soalan-soalan yang disediakan seperti ;

* Adakah pengguna boleh menggunakan aplikasi tanpa bantuan dan arahan?
* Adakah rekabentuk aplikasi mudah dan ringkas?
* Adakah pengguna mengalami gangguan teknikal semas menggunakan aplikasi ini?
* Adakah pengguna boleh mengawal sepenuhnya fungsi dalam aplikasi ini?
* Adalah laras bahasa atau penggunaan simbol cukup difahami oleh pengguna?

Jika banyak soalan yang dijawab oleh pengguna adalah ‘Ya’ ia boleh dikatakan sebagai kebolehgunaan telah tercapai. Pengujian ini penting untuk mengukur tahap kepuasan pengguna dan mengurangkan kos sokongan dan juga dokumentasi.

Pengujian kebolehgunaan boleh dilakukan untuk menguji fungsi dan penerimaan pengguna terhadap antaramuka. Pengujian antaramuka perlu mengambilkira 5 prinsip asas rekabentuk interaksi iaitu konsistensi, kebolehan membuat pemerhatian, boleh dipelajari, kebolehan untuk menjangka dan maklumbalas.

* **Konsistensi**

Konsistensi merujuk kepada kedudukan setiap elemen dalam setiap halaman. Semua elemen perlu kekal pada kedudukan yang sama supaya pengalaman pengguna tidak terganggu untuk menggunakan aplikasi tersebut. Sebagai contoh apabila merujuk kepada papan cerita kajian kes yang diberi tadi, kedudukan untuk KELUAR sistem adalah sama bagi kedua-dua modul dan akan konsisten pada setiap halaman yang berkaitan.

* **Kebolehan membuat pemerhatian**

Kebolehan membuat pemerhatian merujuk kepada sejauh mana pengguna berasa selesa dan selamat untuk menggunakan sebarang elemen tanpa ragu-ragu. Contohnya elemen untuk menerima dan menolak permohonan oleh pensyarah jelas dan mesejnya mudah sampai kepada pengguna tersebut. Oleh yang demikian, pengguna dapat menggunakan aplikasi tanpa sebarang keraguan.

* **Boleh dipelajari**

Boleh dipelajari penting bagi pengalaman pengguna buat kali pertama menggunakan aplikasi. Sejauh mana pengguna boleh mengingati cara untuk mengendalikan aplikasi tersebut perlu diambilkira.

* **Kebolehan untuk menjangka**

Kebolehan untuk menjangka pula ialah pengguna dapat menjangka apa yang akan berlaku apabila menggunakan sebarang elemen dalam aplikasi. Sebagai contoh apabila pengguna memilih fungsi untuk Cadang Tajuk, pengguna dapat menjangka untuk mengisi sesuatu bagi mencadangkan tajuk projek akhir. Pengguna akan dibawa ke halaman untuk memasukkan tajuk Projek Tahun Akhir yang akan dipaparkan dalam senarai yang akan dipohon oleh pelajar nanti.

* **Maklumbalas**

Maklumbalas bertujuan untuk melakukan penambahbaikan pada rekabentuk interaksi yang dihasilkan.

**KAEDAH PENILAIAN**

Antara kaedah penilaian yang boleh digunakan dalam menjalankan pengujian sistem ialah soal selidik. Soal selidik boleh dibangunkan untuk menilaian kebolehgunaan produk berdasarkan prinsip asas rekabentuk di atas. Pilihan jawapan dalam soal selidik boleh ditampilkan dalam dua jenis iaitu bentuk Skala Likert ataupun bentuk Skala Guttmen.

Skala Likert digunakan untuk melihat sejauh mana persetujuan pengguna terhadap produk yang dibangunkan. Jenis ini menampilkan skala yang mewakili maksud seperti

5 = Amat setuju

4 = Setuju

3 = Tidak setuju

2 = Sangat tidak setuju

1 = Tidak pasti

Skala Guttman pula bertujuan untuk mendapatkan data yang lebih jelas, tegas dan konsisten iaitu menampilkan pilihan ‘Ya’ yang mewakili setuju dan ‘Tidak’ untuk tidak atau kurang bersetuju.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama produk yang dinilai :**  **Kategori :**  **Versi (\*jika ada) :**  **Fungsi utama :**  **Tarikh :** | | Sistem Pengurusan Projek Tahun Akhir  Sistem Pengurusan Maklumat  -  Memohon tajuk projek dan menyemak permohonan  12 Disember 2015 | |
| **Prinsip asas** | **Penilaian** | **Tandakan (/) jika Ya dan (x) jika Tidak** | |
| **Ya**  **(/)** | **Tidak**  **(x)** |
| **Konsisten** | Adakah fungsi navigasi sentiasa berada di sebelah kiri? |  |  |
| **Kebolehan membuat pemerhatian** | Nama atau ikon yang digunakan pada butang atau navigasi aman mudah untuk difahami. |  |  |
| **Boleh dipelajari** | Saya mudah untuk memahami sistem ini tanpa perlu diajar berulang kali. |  |  |
| **Kebolehan untuk menjangka** | Apabila menekan sesuatu butang atau hyperteks, saya boleh menjangka apa akan dipaparkan selepas itu. |  |  |
| **Maklum balas** | Adakah anda selesa dengan penggunaan hyperteks pada fungsi manipulasi senarai seperti PADAM/ BATAL/ TERIMA/TOLAK? |  |  |

Jadual 1:Contoh Borang Soal Selidik Untuk Menilai Kebolehgunaan Produk

Daripada hasil soal selidik, rumusan perlu dilakukan bagi mengetahui keputusan pengujian.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Prinsip asas** | **Penilaian** | **Jumlah** | | **Peratus (%)** | |
| **Ya**  **(/)** | **Tidak**  **(x)** | **Ya**  **(/)** | **Tidak**  **(x)** |
| **Konsisten** | Adakah fungsi navigasi sentiasa berada di sebelah kiri? | **20** | **0** | **100** | **0** |
| **Kebolehan membuat pemerhatian** | Nama atau ikon yang digunakan pada butang atau navigasi aman mudah untuk difahami. | **18** | **2** | **90** | **10** |
| **Boleh dipelajari** | Saya mudah untuk memahami sistem ini tanpa perlu diajar berulang kali. | **19** | **1** | **95** | **5** |
| **Kebolehan untuk menjangka** | Apabila menekan sesuatu butang atau hyperteks, saya boleh menjangka apa akan dipaparkan selepas itu. | **19** | **1** | **95** | **5** |
| **Maklum balas** | Adakah anda selesa dengan penggunaan hyperteks pada fungsi manipulasi senarai seperti PADAM/ BATAL/ TERIMA/TOLAK? | **9** | **11** | **45** | **55** |

Jadual 2:Contoh Rumusan Borang Soal Selidik Untuk Menilai Kebolehgunaan Produk

Bagi pengujian paparan dan rekabentuk, penilaian kuantitatif diperlukan bagi mengukur keberkesanan sesuatu produk. Kaedah yang sesuai ialah temubual dan juga soal selidik.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tajuk Sistem** | Sistem Pengurusan Projek Tahun Akhir | | |
| **Objektif Sistem** | Membangunkan sistem permohonan dan semakan tajuk projek tahun akhir dia atas talian. | | |
| **Sasaran** | Pensyarah dan pelajar semester 7 | | |
| **Kategori Pengguna** | Pensyarah/Pelajar | | |
| Tandakan (/) sekiranya anda setuju dan tandakan (x) sekiranya Tidak Setuju | | | |
| **Kategori** | **Kriteria** | **Ya (/)** | **Tidak (x)** |
| Rekabentuk skrin | 1. Pemilihan warna dan ikon yang sesuai dan menarik |  |  |
| 1. Tidak mengandungi kesalahan ejaan |  |  |
| 1. Laras bahasa yang digunakan mudah difahami |  |  |
| Interaktiviti pengguna | 1. Fungsi Terima/Tolak/Sah/Batal berfungsi dengan baik |  |  |
| 1. Paparan tajuk yang disahkan disenaraikan pada tempat yang betul |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kategori** | **Kriteria** | **Ya (/)** | **Tidak (x)** | **Ya (/)**  **%** | **Tidak (x) %** |
| Rekabentuk skrin | 1. Pemilihan warna dan ikon yang sesuai dan menarik | 25 | 5 | 83 | 17 |
| 1. Tidak mengandungi kesalahan ejaan | 30 | 0 | 100 | 0 |
| 1. Laras bahasa yang digunakan mudah difahami | 30 | 0 | 100 | 0 |
| Interaktiviti pengguna | 1. Fungsi Terima/Tolak/Sah/Batal berfungsi dengan baik | 30 | 0 | 100 | 0 |
| 1. Paparan tajuk yang disahkan disenaraikan pada tempat yang betul | 30 | 0 | 100 | 0 |
| **Purata** | | | | **97** | **3** |

**Jadual 3:Rumusan Penilaian Pengguna Terhadap Paparan Dan Rekabentuk Skrin Sistem Pengurusan Projek Tahun Akhir**

Rajah 5:Contoh Rumusan Yang Membuktikan Antaramuka Pengguna Yang Dibangunkan Telah Mematuhi Kesesuaian Terhadap Pengalaman Pengguna

Pengujian unit dan kebolehgunaan boleh dilaksanakan secara lebih formal apabila melibatkan produk yang komersial dan produk yang dibangunkan kepada sesebuah organisasi yang besar dan penggunaan sistemnya lebih meluas. Hanya boleh dilaksanakan pada aplikasi prototaip yang sedang menanti masa untuk pemasangannya. Pengujian dilakukan pada sesi *one-to-one* terhadap pengguna yang betul-betul menggunakan perisian tersebut. Pengujian ini paling teliti dan mengambil kira pelbagai komponen seperti penggunaan setiap butang arahan dan navigasi dalam sistem tersebut. Pengguna mungkin akan diberikan satu tugasan untuk diselesaikan dengan menggunakan sistem tersebut tanpa bantuan mana-mana kumpulan pembangun. Sesi akan dirakam dengan perisian tertentu bagi memastikan apa yang pengguna betul-betul telah laksanakan. Jika antaramuka pengguna tidak menepati dan sukar difahami, pengguna mungkin akan gagal menyiapkan tugasan dan sistem akan dianggap sebagai gagal.

* **PROSEDUR PENYAHPIJATAN**

Terdapat 12 prosedur untuk menghasilkan laporan pengujian dan pepijat yang efektif. Prosedur ini merangkumi apa yang perlu ada didalam laporan pengujian. Prosedur yang dimaksudkan adalah:

1. Tujuan penghasilan laporan pengujian

Keterangan ringkas mengenai objektif penghasilan laporan. Contohnya: ***“Dokumen ini menerangkan pelbagai aktiviti pengujian sistem yang dilakukakan keatas Sistem Pengurusan Pelajar”.***

1. Pengenalan kepada sistem yang diuji

***“Sistem Pengurusan Pelajar ada sistem berasaskan web. Semua maklumat pelajar akan dipaparkan kedalam sistem seperti maklumat peribadi, maklumat kewangan, maklumat pendaftaran pelajaran, maklumat peperiksaan dan maklumat aktiviti. Ada beberapa modul yang ada iaitu modul Pelajar, Kewangan, Subjek dan Aktiviti”.***

1. Skop pengujian

Skop pengujian terbahagi kepada 3 iaitu *in-scope, out-of-scope* dan item yang tidak diuji. *In-scope* bermaksud semua kefungsian sistem perlu diuji bagi memastikan ia memenuhi keperluan pelanggan. *Out-of-scope* pula menerangkan bahagian sistem yang tidak diuji. Contohnya prestasi sistem tidak akan diuji. Item yang tidak diuji pula contohnya Sistem Maklumat Pelajar memerlukan *third-party system* untuk melaksanakan transaksi yuran iaitu Sistem Bendahari. Sistem Bendahari tidak akan diuji kerana sistem itu adalah sistem luaran.

1. Metriks

Metrik akan menjadikan keputusan pengujian akan lebih difahami. Contohnya pada Rajah 3.– Rajah 5

|  |  |  |  |
| --- | --- | --- | --- |
| **Kes Pengujian** | **Kes kajian yang diuji** | **Kes kajian Berjaya** | **Kes kajian gagal** |
| 80 | 75 | 70 | 5 |

**Rajah 6:** Jumlah Kes pengujian yang berjaya/gagal

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Kritikal | Major | Medium | Low | Jumlah |
| Tutup | 25 | 15 | 20 | 0 | 60 |
| Buka | 0 | 0 | 0 | 5 | 5 |

**Rajah 7:** Jumlah Kes pengujian yang gagal

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Pelajar** | **Kewangan** | **Subjek** | **Aktiviti** | **Jumlah** |
| **Kritikal** | 6 | 7 | 8 | 5 | 26 |
| **Major** | 4 | 5 | 2 | 4 | 15 |
| **Medium** | 6 | 8 | 2 | 4 | 20 |
| **Low** | 1 | 2 | 1 | 1 | 5 |
| **Jumlah** | 17 | 22 | 13 | 14 | 66 |

**Rajah 8:** Jumlah semua kegagalan kes pengujian

1. Jenis-jenis pengujian yang dilakukan

Perlu dinyatakan jenis-jenis pengujian yang telah dijalankan seperti *smoke* dan *sanity testing, regression testing* dan *Alpha testing.*

1. Persekitaran pengujian

Menyatakan dengan jelas persekitaran pengujian dimana pengujian sistem dilakukan. Contohnya Rajah 6.

|  |  |
| --- | --- |
| URL | http://smp.edu.my |
| IP pelayan | 192.168.133.22 |
| Pangkalan data | Oracle 12g |
| HP QC/ALM | 192.168.137.22 |

**Rajah 9:** Persekitaran Pengujian

1. Maklumbalas daripada hasil pengujian

Menerangkan isu-isu yang kritikal semasa proses pengujian dilakukan dan cara penyelesaiannya. Maklumbalas ini perlu didokumenkan secara formal agar dapat membantu penguji semasa proses pengujian seterusnya dengan mengelakkan melakukan kesalahan yang sama.

1. Cadangan penambahbaikkan

Cadangan penambahbaikkan dari segi proses pengujian. Sebagai contoh, penguji boleh memberi cadangan untuk menggunakan *tools* bagi menghasilkan kes pengujian secara automatik.

1. Praktik yang efektif bagi pengujian

Penguji mendokumenkan praktik pengujian yang efektif berdasarkan pengalamannya melakukan proses pengujian.

1. Ciri-ciri penamatan pengujian sistem

Bahagian ini sangat penting bagi menentukan bila pengujian sistem akan ditamatkan. Ini kerana, adalah mustahil untuk memastikan sistem 100% bebas daripada pepijat. Antara ciri-ciri yang boleh diambil kira adalah :

i) Semua kes pengujian telah diuji

ii) Semua pepijat yang kritikal telah dibuat penambahbaikkan.

1. *Sign off*

Perjanjian diantara penguji bagi menentukan samaada proses pengujian perlu ditamatkan atau tidak. Ianya banyak bergantung kepada ciri-ciri penamatan pengujian sistem. Hal ini perlu mendapat persetujuan kepada semua penguji.

1. Definisi dan akronim yang digunakan

Mendokumenkan definisi dan akronim untuk dirujuk pada masa hadapan.

**HALAMAN RALAT TERSUAI**

Tujuan Halaman ralat tersuai dibina oleh pembangun sesebuah sistem adalah untuk menunjukkan pengunjung laman web tesebut terdapat kesalahan /ralat pada page yang dikunjungi atau diminta. Halaman ralat tersuai juga boleh berlaku apabila terdapat kegagalan pada sesuatu sistem semasa beroperasi. Ia berfungsi dengan mana-mana tema apabila ia diaktifkan pada blog anda. Pemilik laman web boleh menyesuaikan *404 Not Found* dan halaman ralat lain di laman web mereka.

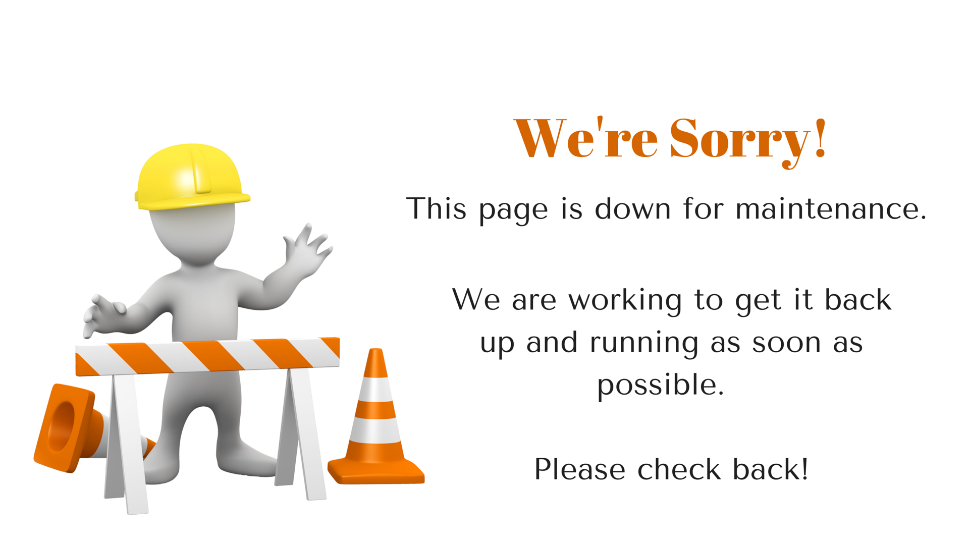
Sebagai contoh, di sini di *How-To Geek* mempunyai ralat khas *404 Page Not Found* yang diilhami oleh permainan Mario klasik. Kesalahan ini bermaksud perkara yang sama, tetapi disesuaikan untuk menjadi lebih mesra dan membantu mencari apa yang anda cari. Contohnya Rajah 7.

Rajah 10: Halaman ralat tersuai

**JENIS RALAT TERSUAI**

1. Sedang dalam Penyelenggaraan (*Under Maintenance)*

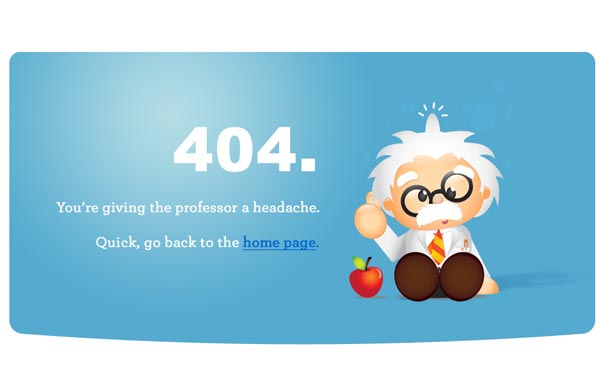
Halaman penyelenggaraan adalah pemegang tempat sementara apabila tapak atau aplikasi perlu diambil di luar talian untuk dikemas kini, dijadikan sandaran atau penyelenggaraan. Ia memberitahu pengguna bahawa laman web yang cuba di capai buat sementara tidak tersedia kerana dalam proses membaik pulih. Ia bukan adegan jenayah, hanya zon pembinaan sementara. Contohnya Rajah 8.



Rajah 11: Sedang dalam Penyelenggaraan

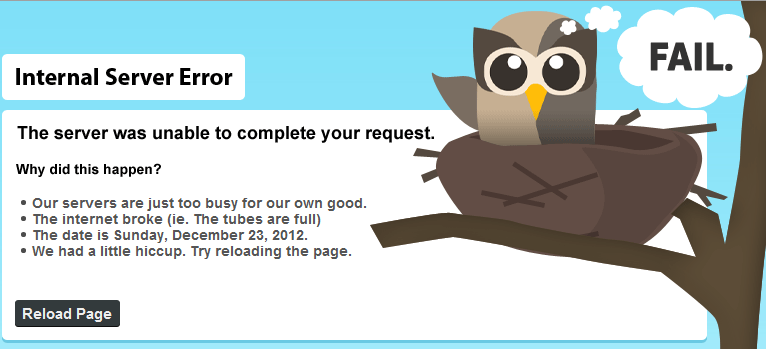
1. File tidak dijumpai *(file not found)*

Anda mungkin melihat pelbagai mesej pelayan web apabila mengakses laman web. Yang paling biasa ialah "404 Not Found," yang bermaksud anda cuba mengakses halaman yang tidak wujud. Sama ada halaman web telah dialih keluar atau anda sedang menaip di alamat dan salah taipkannya. Mesej ralat ini dijana oleh pelayan web jauh dan dihantar ke penyemak imbas anda. Jika anda melihatnya, semak semula alamat halaman web yang anda taipkan. Sekiranya anda mengklik pautan, pautan itu salah - atau halaman yang ditunjukkan olehnya telah dialih keluar. Contohnya Rajah 9.

Rajah 12: Fail tidak dijumpai

1. Rangkaian hilang (Network lost)

Ada sesuatu yang tidak kena pada pelayan web dan ia tidak dapat memproses permintaan atau kemas kini anda.

Rajah 13: Rangkaian hilang

**ALAT PENYAHPIJATAN**

Alat penyahpijatan digunakan untuk empat sebab terutamanya:

* Ujian pepijat. Kelemahan dengan mudah boleh dieksploitasi jika terdapat 'kesilapan' dalam kod anda.
* Menemui sumber pepijat itu
* Mampu mengasingkan pepijat dan kesilapan. Cuba mencari sumber apa yang menyebabkan pepijat.
* Menetapkan dan 'menampal' kesilapan atau logik yang menyebabkan kelemahan yang berpotensi

Antara perisian yang digunakan untuk penyahpijatan

* Advanced Debugger – Standard Penyahpijatan UNIX
* AQtime — profile dan memori/sumber penyahpijatan untuk *Windows*
* CodeView — adalah penyahpijatan untuk platform DOS
* DBG — PHP Debugger dan Profiler
* DEBUG — penyahpijatan terbina dalam DOS dan Microsoft Window
* WinDbg — penyahpijatan pelbagai guna untuk Windows
* GNU Debugger – penyahpijatan mudah alih yang berjalan pada sistem Unix

dan kerja untuk bahasa pengaturcaraan termasuk Ada, C,

C++, Objective-C, Free Pascal, Fortran, Go and Java

**MASA PUSING BALIK *(TURNAROUND TIME)***

Masa pusingan (TAT) bermaksud jumlah masa yang diambil untuk memenuhi permintaan terhadap sesuatu proses. Dalam pengkomputeran, masa pusingan adalah jumlah masa yang diambil antara penyerahan program / proses / tugas untuk pelaksanaan dan pengembalian output lengkap kepada pelanggan / pengguna. Ia mungkin berbeza-beza untuk pelbagai bahasa pengaturcaraan bergantung kepada perisian atau program.

**JENIS-JENIS PENGUJIAN**

1. Pengujian Kefungsian

Ujian Kefungsian merupakan proses pengujian sistem/perisian yang digunakan dalam pembangunan perisian di mana perisian diuji untuk memastikan ia memenuhi semua keperluan.

Ujian kefungsian dilaksanakan dengan cara menyemak sistem/perisian untuk memastikan bahawa ia mempunyai semua fungsi yang diperlukan yang ditentukan dalam keperluan fungsinya.

1. Pengujian Fail

Pengujian dilaksanakan berdasarkan saiz fail dan juga format pada fail tersebut samada disokong oleh sistem tersebut ataupun tidak.

Contohnya memuat naik fail dengan mengambil kira saiz dan format fail.

1. Pengujian Data Integriti

Mengesahkan data dalam pangkalan data yang tepat dan berfungsi seperti yang diharapkan. Data yang dimasukkan perlu diselaraskan dengan *data dictionary.*

Merujuk kepada proses manual atau automatik yang digunakan oleh pentadbir pangkalan data untuk memastikan ketepatan, kualiti dan fungsi data yang disimpan dalam pangkalan data.

**SOALAN/***QUESTION***:**

1.Jelaskan definisi bagi Array.

2.Terdapat dua jenis array iaitu…..

2.Terangkan prosedur penyahpijatan.

3.Berikan jenis-jenis ralat tersuai.

4.Terangkan jenis-jenis pengujian.

**RUJUKAN/***REFERENCE* **:**

1. Keith Cooper, Linda Torczon(2011), Engineering a Compiler (2nd Edition), Morgan Kaufmann, ISBN: 978-0120884780
2. Todd Zaki Warfel (2009), Prototyping: A Practitioner's Guide, Louis Rosenfeld Media, LLC, ISBN I-933820-21-7
3. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
4. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
5. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946