

BAB III EKSPERIMEN AWAL

Pada bab ini akan dipaparkan rangkaian uji coba awal penggunaan *software* Rational yang telah dilakukan oleh penulis dalam proses pembelajaran. Uji coba yang telah dilakukan untuk mempelajari fungsi-fungsi dasar yang ada pada *software* Rational. Uji coba dilakukan pada sistem ClassicsCD.com, yaitu sebuah aplikasi toko *online* yang menjual CD musik. Sistem tersebut merupakan contoh sistem yang telah disediakan pada Rational Suite. Bab ini akan memaparkan beberapa hal cukup rinci agar pembaca bisa mengevaluasi dan mengulang kembali uji coba yang dilakukan penulis.

3.1 Instalasi *software* ClassisCD.com

Instalasi dilakukan dengan mengeksekusi *file* RationalSuiteTutorialv2003.exe. Proses instalasi dilakukan dengan mengikuti instruksi yang tampil. Instalasi harus dilakukan pada direktori C:\ pada Windows. Setelah instalasi selesai, maka akan ada *folder*Classics pada direktori C:\. Di dalam *folder* tersebut berisi empat *folder*, yaitu *folder* ClassicsApp, ClassicsCD.com_sites, Projects dan Web Publish. Hanya ada dua *folder* yang akan digunakan untuk uji coba, yaitu *folder* ClassicsCD_com_sites dan Projects. *Folder* berisi source code aplikasi ClassisCD.com, sedangkan *folder*Projects berisi *file* yang berkaitan dengan *software* Rational Suite.

3.2 Modifikasi *source code*

Source code index.html pada aplikasi ClassicsCD.com menggunakan *tag* xml yang telah divalidasi. Namun, ternyata kode yang ada pada dua baris awal tidak valid. Hal tersebut menyebabkan Internet Explorer hanya dapat membuka *file* index.html tanpa dapat menjalankan fungsi yang ada didalamnya. Hal ini hanya berlaku pada Internet Explorer. Oleh karena itu, source code yang ada harus sedikit dimodifikasi. Modifikasi hanya dilakukan dengan menghapus dua baris pertama dari setiap *file* index.html yang ada pada *folder*ClassicsCD_com_sites. Modifikasi ini ditujukan agar aplikasi ClassicsCD.com dapat berjalan dengan baik

pada Internet Explorer yang ada pada Windows.

3.3 Registrasi aplikasi pada *software* Rational

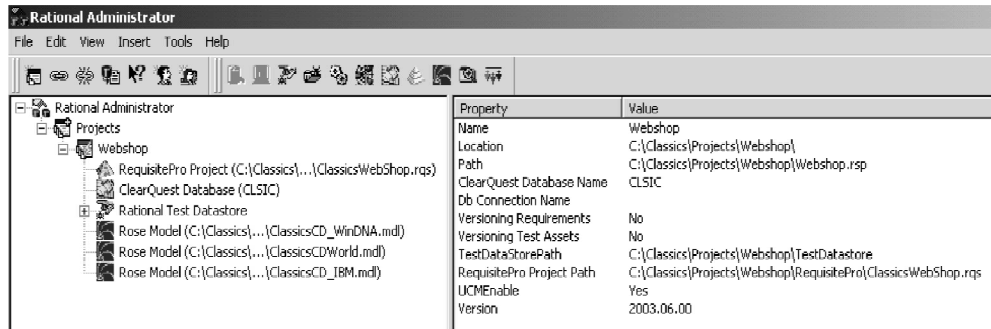
Untuk dapat menggunakan *software* Rational, sebelumnya kita harus meregistrasi aplikasi yang akan diuji coba [RATU03]. Registrasi dilakukan dengan beberapa langkah, yaitu:

- Menjalankan aplikasi Rational Administrator dengan menklik Start>Programs>Rational *Software*> Rational Administrator.
- Pada bagian sebelah kiri tampilan Rational Administrator, klik kanan Project dan pilih Registering Existing Project.



Gambar 3-1. Registrasi pada Rational Administrator

- Setelah itu akan muncul *dialog box* Select Rational Project, arahkan ke direktori C:\Classics\Projects\Webshop.
- Pilih *file* Webshop.rsp dan klik Open. Rational Administrator akan menampilkan proyek Webshop yang telah diregistrasi. Proyek ini akan digunakan untuk melakukan seluruh uji coba.
- Pada bagian tampilan Rational Administrator klik kanan Webshop dan klik Connect. Seluruh *file* dan informasi pengembangan yang berhubungan dengan proyek Webshop akan tampil. Untuk melakukan tahap berikutnya Rational Administrator harus tetap berjalan.



Gambar 2-2. Properti dari proyek Webshop

3.4 Perencanaan uji coba

Rational TestManager dapat digunakan untuk melakukan perencanaan uji coba. Semua perencanaan uji coba aplikasi ClassicsD.com sudah tersedia pada proyek Webshop. Pada tahap ini akan ditunjukkan bagaimana menggunakan fungsi-fungsi yang ada pada TestManager. Langkah-langkah yang akan dilakukan adalah:

- Menjalankan aplikasi Rational TestManager dengan cara klik Start>Programs>Rational *Software*>Rational TestManager.
- Setelah muncul Rational Test Login *dialog box*, isi *box* yang kosong sebagai berikut:
 - Pada box *User Name* dan *Password* ketik pat.
 - Pastikan pada box *Project* menampilkan Webshop.
 - Pastikan pada box *Location* menampilkan C:\Classics\Projects\Webshop\Webshop.rsp
 - Klik OK.

TestManager akan terbuka dan menampilkan proyek Webshop.

3.5 Mempelajari stuktur dari ClassicsCD *Test Plan*

Sebuah *Test Plan* dalam TestManager mengandung informasi tentang tujuan dan target dari sebuah proyek Rational, dan strategi untuk mengimplemantasikan dan menjalankan *testing*. Sebuah proyek dapat terdiri dari satu atau lebih *Test Plan*. Sebuah *Test Plan* dapat terdiri dari berbagai properti seperti nama *Test Plan*, konfigurasi *Test Plan*, dan sebuah frame waktu. Sebuah

laporan dapat dibuat berdasarkan property yang ada pada *test plan*. Sebuah *test plan* berisi banyak *folder test case* yang berisi berbagai *test case*.

Pada proyek ClassicsCD.com sudah tersedia *Test Plan* yang akan digunakan untuk melakukan pengujian. Untuk memahami struktur dari *Test Plan* ClassicsCD.com dilakukan beberapa langkah sebagai berikut:

- Buka *folder test plan* yang ada pada *tab Planning* di bagian *Test Asset Workspace* dan klik dua kali pada ClassicsCD. Setelah itu akan muncul *folder test case* dari proyek Webshop pada bagian sebelah kanan. *Folder-folder test case* tertata berdasarkan *use case*.
- Buka *folder test case* untuk *use case* Purchase CD. Akan muncul dua *folder*, yaitu *Functionality* dan UI. Semua *test script* yang berhubungan dengan perilaku sistem (fungsionalitas) akan ada pada *folder Functionality*, sedangkan representasi *website* dari fungsionalitas tersebut akan ada pada *folder user interface* (UI).
- Setelah itu buka *folder test case* UI dan *folder* Order Confirmation Page yang ada di dalamnya. *Folder* tersebut akan kosong, dan akan dibuat *test script*-nya.

Sebuah *test case* menggambarkan bagian yang bisa diuji dan variabel perilaku dari sebuah sistem. Sebuah *test plan* dapat juga menggambarkan seluas apa area dari aplikasi yang akan diuji. Untuk mengetahui bagaimana membuat sebuah *test case*, berikut adalah beberapa langkah-langkahnya:

- Pada bagian kanan TestManager, klik kanan *folder* Order Confirmation Page dan pilih menu *Insert Test case*.
- Akan muncul sebuah tampilan *dialog box*. Pada *box* Name ketik *Display Estimated Ship Date*. Pada *box* Description ketik *Warehouse sistem gets order and responds with estimated ship date*. Pastikan pada *box* Owner terpilih *user pat*.
- Klik OK. *Test case Display Estimated Ship Date* akan muncul pada susunan *test plan* yang ada.

3.6 Functional testing

Functional testing membantu dalam menentukan apakah sistem berjalan

seperti seharusnya. Cara konvensional untuk menguji coba perilaku sistem adalah dengan menggunakan tampilan GUI dari sistem untuk memvalidasi bahwa sistem merespon *input* yang diberikan dengan baik. Uji coba yang dilakukan dapat berfokus pada operasi dan penampilan dari objek GUI. TestManager menyediakan dukungan untuk mengimplementasikan dan menjalankan *functional test* yang dibuat pada Robot.

Dengan menggunakan Robot, sebuah *test script* dapat dibuat dengan merekam setiap aktivitas sistem. Setiap aktivitas sistem dapat direkam menjadi sebuah *test script* ataupun beberapa *test script* yang spesifik. Dengan Robot, dapat juga dibuat sebuah *test script* modul singkat yang dapat dikombinasikan dan dirangkai ke dalam sebuah pengujian dalam TestManager. Dengan teknik tersebut, *test script* dapat digunakan kembali pada pengujian yang berbeda. Contoh *test script* yang dapat dibuat adalah:

- *Log in* ke sistem.
- Memilih barang untuk dibeli.
- Melakukan transaksi pembelian.
- *Log out* pada sistem.

Script yang ada dapat dijalankan secara individual maupun secara bersamaan dengan mengkombinasikan *test script* tersebut ke dalam sebuah rangkaian pengujian.

3.7 Penggunaan Robot

Pada bagian ini, *test script* yang telah digunakan untuk merekam aplikasi ClassicsCD.com akan digunakan kembali untuk merekam *test script* untuk *test case* yang telah dibuat sebelumnya [RATU03].

Untuk merekam menggunakan Robot dijabarkan sebagai berikut:

- Jalankan Robot dengan mengklik Tools>Rational Test>Rational Robot.
- Pada tampilan Robot, klik Tools>GUI Playback Options.
- Pada box GUI Playback Options pilih *tab Log*. Pastikan beberapa hal di bawah ini telah terpilih:
 - *Output playback results to log*.
 - *View log after playback*.

- *Specify log information at playback.*
- Klik OK.
- Tutup Robot dengan memilih menu *File>Exit*.

3.8 Penggunaan *Test script*

Pada uji coba ini, *script* dapat direkam dan dijalankan kembali hanya jika Microsoft Internet Explorer telah terinstal pada komputer. Berikut adalah langkah untuk menjalankan *script* melakukan transaksi penjualan:

- Pilih *tab Execution* pada tampilan *Test Asset Workspace* pada TestManager. Buka rangkaian tes yang ada dan klik dua kali pada ClassicsCD Buy Beethoven Bach v1a. *Test script* untuk rangkaian tes tersebut akan muncul pada bagian kanan TestManager.
- Pada bagian *Test Asset Workspace*, klik kanan ClassicsCD Buy Beethoven Bach v1a dan pilih menu *Run*. Sebuah *dialog box* dari TestManager *Run Suite* akan muncul. Pastikan pada kolom *Build list* telah terpilih *Build 1*. Bila belum terpilih, klik *change* dan pilih *Build 1*.
- Klik OK pada *dialog box* TestManager *Run Suite*. Jika TestManager meminta untuk menu menulis kembali sebuah test log, klik No. Selanjutnya proses uji coba akan berjalan. Jangan berinteraksi dengan aplikasi ClassicsCD.com selama proses berlangsung.

Script yang dijalankan melakukan transaksi pemilihan dua barang dan pembayaran. Pada titik tertentu *script* akan membandingkan nilai yang ada pada aplikasi yang berjalan dengan sebuah nilai *baseline*. Ketika *script* telah selesai dijalankan, TestManager akan menampilkan hasil dari tes yang telah dilakukan pada sebuah *Test Log Window*. Semua perbandingan yang telah dilakukan (*verification point*) akan menampilkan status *pass*.

Pada tampilan *Test Log*, hasil pengujian dapat dipelajari lebih lanjut dengan membuka setiap hasil yang ada. Pada kasus tertentu akan muncul status Warning yang menyatakan *Unexpected Active Window*. Status tersebut berarti bahwa selama proses, aplikasi lain telah tampil atau berjalan.

- Tutup tampilan *Test Log*.

3.9 Merekam sebuah *script*

Pada bagian ini akan dipaparkan bagaimana merekam sebuah *script* dengan menggunakan Robot. Contoh yang dibuat akan merekam pada tampilan ClassicsCD Buy Bethoven Bach v1a. Berikut adalah langkah-langkah untuk merekam:

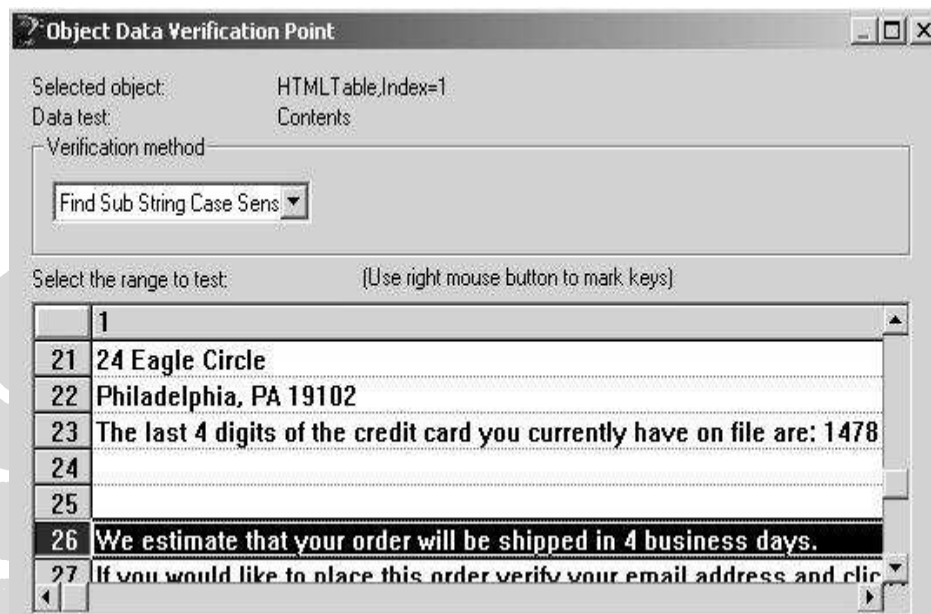
- Buka *website* ClassicsCD.com dengan cara membuka direktori C:\Classics\ClassicsCD_com sites\v1a dan buka *file* Cashier.htm. *Browser* akan menampilkan langsung halaman *cashier* dari ClassicsCD.com v1a.
- Cari kata *Place Order*. Pastikan bahwa *estimated ship date* tampil.
- Pada TestManager, klik tombol *Record GUI*. Rational Robot akan berjalan dan menampilkan *dialog box Record GUI*.
- Pada *box Name*, ketik *Display Estimated Date* lalu klik OK. Sebuah GUI *Record toolbar* akan segera tampil.
- Buat agar tampilan ClassicCD.com adalah tampilan yang sedang aktif.

3.10 Membuat sebuah *Verification point*

Untuk *test script* yang dibuat di atas, sebuah *Verification point* juga harus dibuat untuk membuat nilai *baseline* untuk *object property* [RATU03]. Ketika *script* yang dibuat dijalankan kembali, Robot akan membandingkan nilai yang ada dengan nilai *baseline* yang telah dibuat. Sebuah *test script* dapat berisi lebih dari satu *Verification point*. Tahap-tahap untuk membuat *Verification point* adalah sebagai berikut:

- Pada GUI *Record toolbar*, klik *insert Verification point*.
- Pilih *Object Data Button*. Sebuah dialog box akan muncul.
- Pada *box Name* ketik *Verify Ship Date*, kemudian klik OK.
- Klik *check box* sehingga *dialog box* akan tertutup secara otomatis ketika memilih sebuah *verification point*.
- Arahkan pointer yang berbentuk tangan pada tampilan ClassicsCD.com yang memiliki kata *We estimate that your order*.
- Sebuah *dialog box Object Data Test* akan muncul. Hal tersebut menunjukkan bahwa content dari HTML *table* telah direkam. Klik OK untuk melihat tulisan yang telah direkam.

- Pada *dialog box Object Data Verification point*:
 - Klik *Find Sub String Case Sensitive* yang ada pada daftar *Verification method*.
 - Pada tampilan *Select the range to test*, klik pada baris 26, sehingga itu adalah satu-satunya baris yang dipilih.
 - Klik OK untuk menutup *dialog box*.



Gambar 3-3. Contoh verifikasi object data

Sekarang sebuah *Verification point* telah dibuat dan proses perekaman *test script* hampir selesai. Ketika *script* tersebut dijalankan, akan memverifikasi bahwa kata-kata yang telah direkam masih ada.

3.11 Menyelesaikan sesi perekaman

Setelah tampilan yang ingin direkam telah direkam, kita perlu menyelesaikan sesi perekaman terlebih dahulu. Sesi perekaman dapat segera diselesaikan dengan cara:

- Pada halaman Checkout, klik *Place Order*.
- Klik *stop recording* pada GUI Record toolbar. Robot akan menampilkan

script Display Estimated Ship Date yang telah direkam.

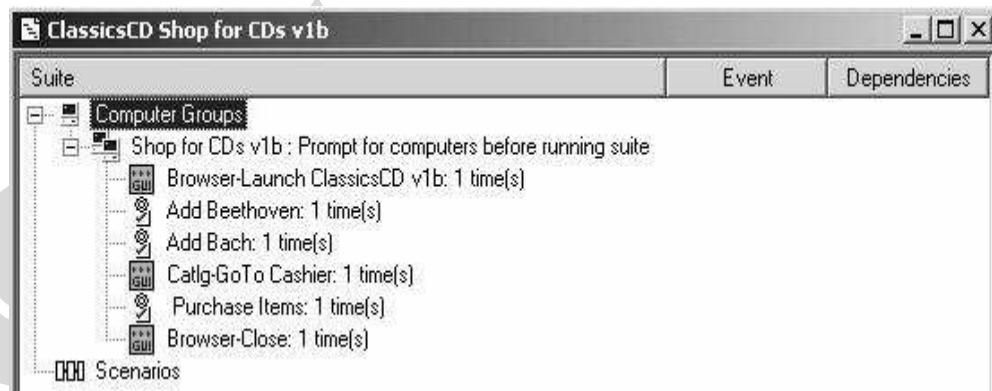
- Tutup Robot dan ClassicsCD.com.

3.12 Menambahkah sebuah *test script* ke sebuah *suite*

Test script yang sudah jadi dapat ditambahkan langsung ke dalam *suite*.

Berikut adalah langkah untuk menambahkan *test script* ke dalam *suite*:

- Klik dua kali pada ClassicsCD Shop for CDs v1b yang ada pada *tab Execution. Test case* dan yang *script* yang ada akan tampil di bagian kanan.



Gambar 3-4. Hirarki Webshop suite

- Klik kanan pada *Purchase Items: 1 time(s)* dan pilih *Insert>Test script*.
- Pada box *Test script* box klik GUI-(Rational Test Datastore).
- Pada kolom Selection pilih *Display Estimated Ship Date*.
- Klik OK.
- Pilih *File>Save*.

3.13 Memasukkan sebuah test sript ke dalam *Test Plan*

Setelah dimasukkan ke dalam *Suite*, *test script* juga harus dimasukkan ke dalam *Test Plan*. Langkah-langkah untuk memasukkannya adalah sebagai berikut:

- Buka *tab Planning* yang ada pada *Test Asset Workspace*. Pilih *Test Plans>ClassicsCD* dan klik dua kali pada ClassicsCD. *Test Plan* ClassicsCD akan muncul pada bagian kanan.
- Pilih *ClassicsCD>Purchase CD UC>UI>Order Confirmation Page>Display Estimated Ship Date*.
- Klik kanan pada *Display Estimated Ship Date* dan pilih *Properties. Dialog*

box Properties akan muncul.

- Pada tab *Implementation*, dibawah *Automated Implementation* klik *Select* dan klik *GUI-(Rational Test Datastore)*.
- Pilih *Display Estimated Ship Date* dan klik *OK*.
- Pada tampilan *Test case Properties* klik *OK*. Sekarang *test script* sudah masuk ke dalam *Test Plan*.

3.14 Menjalankan kembali *script* dengan *New build*

Pada bagian ini akan dilakukan uji coba fase kedua. *Test case* yang akan dijalankan berasal dari aplikasi versi kedua. Langkah-langkah untuk menjalankan *test script* adalah sebagai berikut:

- Buka tab *Execution* yang ada pada *Test Asset Workspace* TestManager. Kemudian klik kanan pada *ClassicsCD Shop for CDs v1b* dan klik *Run*. *Dialog box* TestManager *Run Suite* akan muncul. Perhatikan bahwa *Build 1* tertera pada bagian *Log Information*.
- Klik *Change* pada *Build List* dan pilih *Build 2*. Kemudian klik *OK*.
- Klik *OK* pada *dialog box Run Suite*. Jangan melakukan interupsi selama prose berlangsung. Setelah selesai, akan muncul tampilan *Test Log*.
- Klik tab *Details* untuk melihat hasil pengujian.

3.15 Analisa hasil pengujian

TestManager akan memperlihatkan *Verification point* mana yang berhasil dan gagal. Dengan melihat *test log* yang ada kegagalan dalam pengujian bisa diatasi. Pada pengujian dengan menggunakan *Build 2*, beberapa hasil pengujian akan menunjukkan kegagalan. Pengujian yang gagal dapat ditangani dengan menggunakan TestManger. Ada dua tipe kegagalan *script*, yaitu:

- Kegagalan yang disengaja karena perubahan pada aplikasi. Pada kasus ini, *Verification point* yang gagal dapat diganti dengan *baseline* yang seharusnya.
- *Error* yang sebenarnya dimana *script* gagal karena berbeda dengan *baseline*. *Error* yang ada dapat dilaporkan menggunakan *Rational ClearQuest* yang terintegrasi pada TestManager.

Dengan menggunakan TestManager kegagalan *script* yang disengaja dapat diatasi. Berikut adalah langkah-langkah untuk mengatasi kegagalan disengaja:

- Pilih tab detail pada tampilan Test Log. Kemudian pilih *Computer Start (Shop for CDs v1b(1))>Test Case Start (Purchase Items)> Script Start (Catlg-Purchase Items)*.

Event Type	Result
Suite Start (ClassicsCD Shop for CDs v1b)	Fail
Computer Start (Shop for CDs v1b [1])	Fail
+ Script Start (Browser-Launch ClassicsCD v1b)	Pass
+ TestCase Start (Add Beethoven)	Pass
+ TestCase Start (Add Bach)	Pass
+ Script Start (Catlg-GoTo Cashier)	Pass
- TestCase Start (Purchase Items)	Fail
- Script Start (Catlg-Purchase Items)	Fail
- Verification Point (Page Header - Object ...)	Fail
- Verification Point (Order Summary - Objec...	Fail
- Script End (Catlg-Purchase Items)	Fail
- TestCase End (Purchase Items)	Fail
+ Script Start (Display Estimated Ship Date)	Pass
+ Script Start (Browser-Close)	Pass
Computer End	Fail
Suite End (ClassicsCD Shop for CDs v1b)	Fail

Gambar 3-5. Test log Webshop

- Klik dua kali pada *Verification point (Page Header-Object Data)*. Tampilan *Grid Comparator* akan muncul untuk menunjukkan bahwa page header telah berganti dari *Checkout* menjadi *Cashier*. Hal tersebut sudah direncanakan pada bagian modifikasi UI sebelumnya. Pada kasus ini *baseline* yang ada dapat diganti.
- Untuk mengganti *baseline* klik *File>Replace Baseline With Actual*. Pada *Grid Comparator* akan muncul konfirmasi pergantian *baseline*.
- Klik Yes. *Grid Comparator* akan memperbaharui *baseline* sehingga tidak ada perbedaan.
- Tutup *Grid Comparator*.

Error yang sebenarnya juga dapat dilihat menggunakan TestManager.

Langkah-langkahnya adalah sebagai berikut:

- Pada tampilan *Test Log* klik dua kali pada *Verification point (Order Summary-Object Data)*. *Grid Comparator* akan muncul. Pada baris kedua, *baseline* menunjukkan bahwa *output* yang diharapkan adalah Beethoven

Symphony, sedangkan yang actual adalah Mozart Symphony. Hal tersebut merupakan *Error* yang sebenarnya karena harga CD menjadi berbeda.

- Tutup *Grid Comparator*.

3.16 Melaporkan *Error*

Langkah-langkah untuk melaporkan *error* menggunakan ClearQuest yang terintegrasi dengan TestManager adalah sebagai berikut:

- Klik kanan pada *Verification point* (Oder Summary-Object *Path*) dan pilih Submit Defect. Jika ClearQuest telah terinstal maka akan muncul *dialog box* ClearQuest Login.
- Login dengan menggunakan *username* dan password pat. Pada kolom Database pastikan CLSIC terceklist. Kemudian klik OK. ClearQuest akan terbuka dan secara otomatis memberikan angka pada *Error*.
- Ketik Wrong item in shopping cart pada box *Headline*.
- Klik 1-Critical pada kolom *Severity*.

Submit Defect CLSIC00000130

Iterations & Notes | Unified Change Management | Requirements

Main | **Test Data** | Environment | Attachments

ID: CLSIC00000130 State: Submitted

Headline: Wrong item in shipping cart

Suite Project: Webshop Keywords:

UCM Project: Keywords: ...

Owner: Symptoms:

Priority: Symptoms: ...

Severity: 1-Critical

Customer Priority:

Description:

***** Auto-Generated by TestManager *****

OK

Cancel

Gambar 3-4. Tampilan ClearQuest

- Buka *tab Test Data* dan pastikan TestManager telah memasukkan *Error* yang ada.
- Klik OK.
- Buka tampilan *Test Log* dan *scroll* sampai kolom *Defect* terlihat. *Testing* pada ClassicsCD.com telah selesai dilakukan.