4/6/2013

Supermarket Automation Software

Test Results Document

Siddharth Rakesh 11CS30036

Sachin Kumar 11CS30043

Table of Contents

Black Box Testing

We did error guessing and boundary value analysis on all parameters and variables in a method.

BBT for addEmployee method

This function takes a parameter of employee type (SalesClerk or Supermarket Staff) and sends that object to the server by serializing it and server stores the data of the object in the employee table in the database.

Tester name	Sachin Kumar	Test Date	5 April 2013
Class Name	InventoryDatabase	Method Name	addEmployee
Parameter		Employee details	
First Name		Akash	
Last Name		Hajela	
Post		Sales Clerk	
Salary		10000	
Address		C-98, Railway Colony, Kharagpur	
Phone		8492562228	
Bank Account Number		345324623224	
Email id given exists in the database		Error Message	
Cannot connect to the server		Error message	
No discrepancy		Employee ID and randomly generated password returned.	
Test Case	Database not connected	Email exists?	Email does not exist?
Expected output	Error message	Error message	6 char long empId– 8 char long password
Actual output	Error message	Error message	SC0020 – 78HYCKY0
Bug found?	No	No	No

BBT for change password

An employee is registered by the manager and a randomly generated password is given to him/her which he/she can give to the employee for logging in. But the employee may want to change his/her password.

Tester name	Sachin Kumar	Test Date	5 April 2013	
Class Name	EmployeeDatabase	Method Name	changePassword	
Parameter		Old Password, new Password, Employee Id		
Employee id		SC0020		
Old Password		78HYCKY0		
New Password		haj.ak456		
Employee id given does not exist in the database		Error message		
Cannot connect to the server		Error message		
Wrong old password		Error message		
No discrepancy		Password successfully changed message		
Test Case	Database not connected	Employee ID not found	Wrong old password	No discrepancy
Expected output	Error message	Error message	Error message	Success message
Actual output	Error message	Error message	Error Message	Success message
Bug found?	No	No	No	No

BBT for Login

Any Employee (Sales clerk, supermarket staff or manager) can log in to his/her account by providing his/her employee id and password.

Tester name	Sachin Kumar	Test Date	5 April 2013	
Class Name	EmployeeDatabase	Method Name	Login	
Parameter		Employee Id, password		
Employee id		SC0020		
Password		haj.ak456		
Employee id given does not exist in the database		Error message. LOGIN0		
Cannot connect to the server		Error message. E001		
Wrong password		Error message. LOGIN0		
No discrepancy		Login successful. LOGIN1		
Test Case	Database not connected	Employee ID not found	Wrong password	No discrepancy
Expected output	Error message	Error message	Error message	Success message
Actual output	Error message	Error message	Error Message	Success message
Bug found?	No	No	No	No

BBT for getEmployee

A manager can view details of any employee using this method. It returns an employee object.

Tester name	Sachin Kumar	Test Date	5 April 2013
Class Name	EmployeeDatabase	Method Name	getEmployee
Parameter		Employee Id	
Employee id		SC0020	
Password		haj.ak456	
Employee id given does not exist in the database		Error message. E002	
Cannot connect to the server		Error message. E001	
Employee id found		Employee object returned.	
Test Case	Database not connected	Employee ID not found	No discrepancy
Expected output	Error message	Error message	Employee object returned
Actual output	Error message	Error message	Employee object returned but all the details are not shown.
Bug found?	No	No	Yes

BBT for addProduct

This method adds a new product to the database

Tester name	Sachin Kumar	Test Date	5 April 2013
Class Name	Inventory Database	Method Name	addProduct
Parameter		Product Details	
product id		1234567890	
Product id given does not exist in the database		Error message. E002	
Cannot connect to the server		Error message. E001	
Employee id found		Employee object returned.	
Test Case	Database not connected	Employee ID not found	No discrepancy
Expected output	Error message	Error message	Employee object returned
Actual output	Error message	Error message	Employee object returned
Bug found?	No	No	No

BBT for updateInventory

This method takes the parameters as product ID and quantity to be added/removed and sends requests to the server to update the database.

Tester name	Siddharth Rakesh	Test Date	6 April 2013
Class Name	Inventory Database	Method Name	updateInventory
Parameter		Product Details	
product id		1234567890	
Quantity		100	
Updation type		1	
Product id given does not exist in the database		Error message. E002	
Cannot connect to the server		Error message. E001	
Product id found		Product quantity updated. Success message returned.	
Test Case	Database not connected	Product ID not found	No discrepancy
Expected output	Error message	Error message	Success and quantity updated in the database
Actual output	Error message	Error message	Success
Bug found?	No	No	No

BBT for addBill

This method takes a sales transaction object as an argument and sends requests to the server for storing the transaction bill in the database.

Tester name	Siddharth Rakesh	Test Date	6 April 2013
Class Name	Inventory Database	Method Name	addBill
Parameter		Sales Transaction	
Transaction ID		000000000016	
Cannot connect to the server		Error message. E001	
Bill added to database		Success message.	
Test Case	Database not connected	Transaction ID not found	No discrepancy
Expected output	Error message	Failure message	Success message
Actual output	Error message	Failure message	Success message
Bug found?	No	No	No

BBT for getOverallProfit

This method takes the begin date and end date strings as arguments and returns the overall profit statistics over time in the form of a hashmap.

Tester name	Siddharth Rakesh	Test Date	6 April 2013
Class Name	Inventory Database	Method Name	GetOverallProfit
Parameters		Begin date, End date	
Cannot connect to the server		Error message. E001	
Test Case	Database not connected	Statistics not found	No discrepancy
Expected output	Error message	Null returned, handled	Profit hashmap returned
Actual output	Error message	Null returned, handled	Profit hashmap returned
Bug found?	No	No	No

BBT for getOverallStats

This method takes begin date and end date strings as arguments and returns the overall profit distribution for the various products in the form of hashmap.

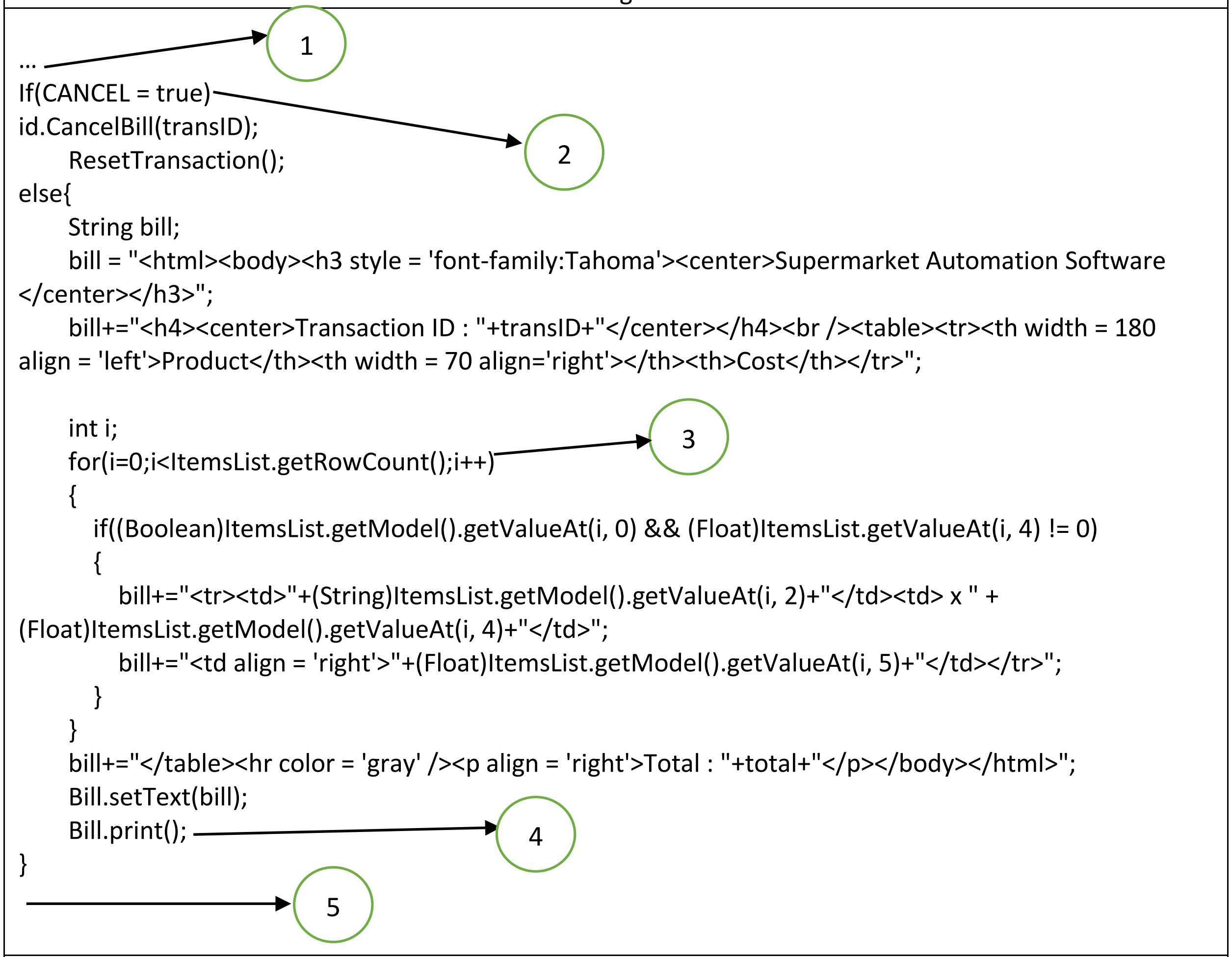
Tester name	Siddharth Rakesh	Test Date	6 April 2013
Class Name	Inventory Database	Method Name	GetOverallStats
Parameters		Begin date, End date	
Cannot connect to the server		Error message. E001	
Test Case	Database not connected	Statistics not found	No discrepancy
Expected output	Error message	Null returned, handled	Distribution hashmap returned
Actual output	Error message	Null returned, handled	Distribution hashmap returned
Bug found?	No	No	No

White Box Testing

In the White Box Testing, we will verify that all the paths in a function are correct through basic path testing. White box tests for different functions are as following:

WBT for testing endTranscation

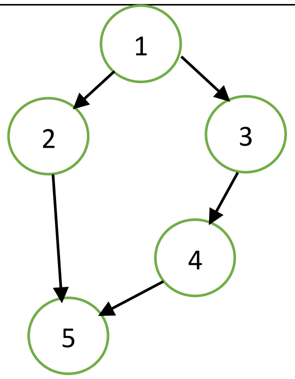
Test ID		Tester Name	Siddharth Rakesh
Class Name	SalesClerkFrame		
Code Segments			



```
…
If(CANCEL = true)
id.CancelBill(transID);
      ResetTransaction();
else{
      String bill;
      bill = "<html><body><h3 style = 'font-family:Tahoma'><center>Supermarket Automation Software
</center></h3>";
      bill+="<h4><center>Transaction ID : "+transID+"</center></h4><br /><table><tr><th width = 180
align = 'left'>Product</th><th width = 70 align='right'></th><th>Cost</th></tr>";

      int i;
      for(i=0;i<ItemsList.getRowCount();i++)
      {
         if((Boolean)ItemsList.getModel().getValueAt(i, 0) && (Float)ItemsList.getValueAt(i, 4) != 0)
         {
            bill+="<tr><td>"+(String)ItemsList.getModel().getValueAt(i, 2)+"</td><td> x " +
(Float)ItemsList.getModel().getValueAt(i, 4)+"</td>";
            bill+="<td align = 'right'>"+(Float)ItemsList.getModel().getValueAt(i, 5)+"</td></tr>";
         }
      }
      bill+="</table><hr color = 'gray' /><p align = 'right'>Total : "+total+"</p></body></html>";
      Bill.setText(bill);
      Bill.print();
}
```

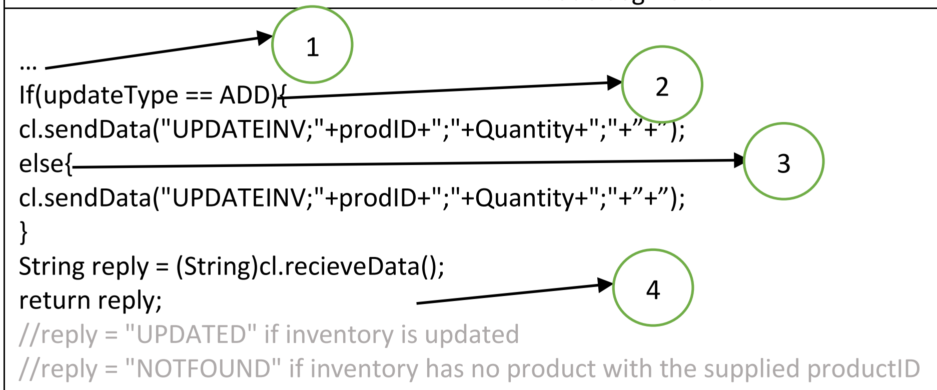
Path Diagram

<table>
<tr><td colspan="2"> </td></tr>
<tr><td>Path 1</td><td>1-2-5</td></tr>
<tr><td>Path 2</td><td>1-3-4-5</td></tr>
<tr><td>Path 1 Expected Result</td><td>Sales transaction Id deleted from the database</td></tr>
<tr><td>Path 2 Expected Result</td><td>Sales transaction complete and sales history gets updated in the database.</td></tr>
<tr><td>Bug found</td><td>If cancel transaction button is pressed multiple times, unexpected exception is thrown and the application hangs.</td></tr>
</table>

WBT for Update Inventory

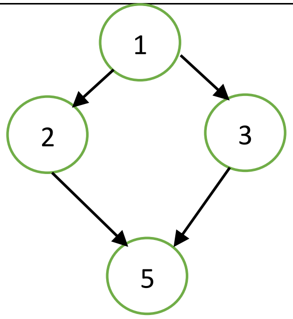
Test ID		Tester Name	Siddharth Rakesh
Class Name	InventoryDatabase		

Code Segments



```
…
If(updateType == ADD){
cl.sendData("UPDATEINV;"+prodID+";"+Quantity+";"+”+”);
else{
cl.sendData("UPDATEINV;"+prodID+";"+Quantity+";"+”+”);
}
String reply = (String)cl.recieveData();
return reply;
//reply = "UPDATED" if inventory is updated
//reply = "NOTFOUND" if inventory has no product with the supplied productID
```

Path Diagram



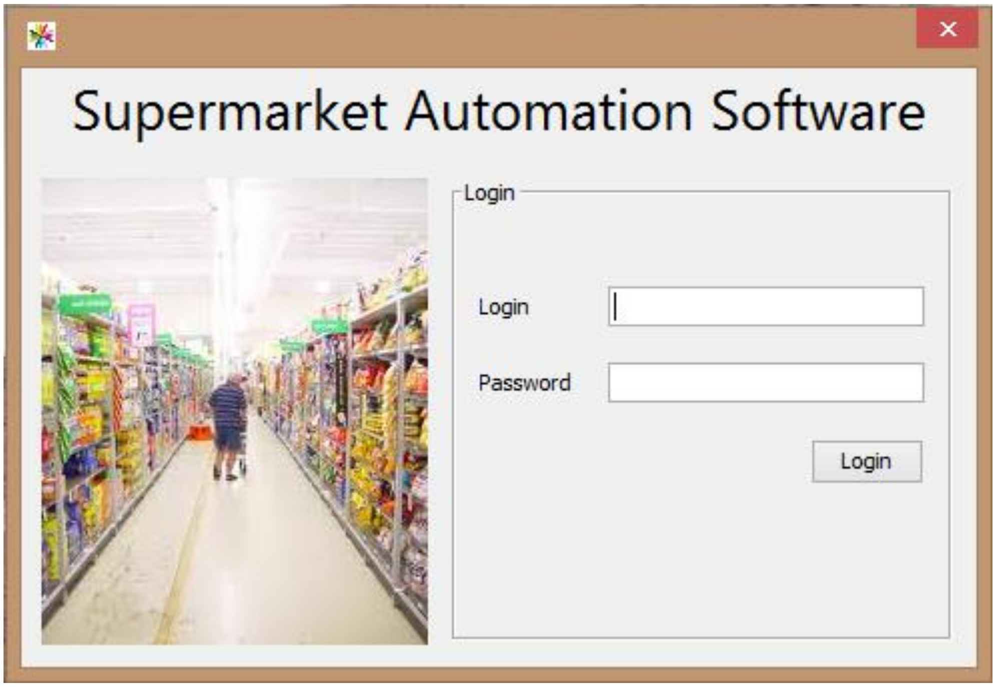
Path 1	1-2-5
Path 2	1-3-5
Path 1 Expected Result	Inventory added to the database
Path 2 Expected Result	Inventory removed to the database.
Bug found	If more than one people remove inventory from various terminals the inventory is not refreshed in the other terminals automatically. If the inventory goes less than zero. Then in some cases database may contain a negative value.

User Interface Testing

To test the user interface, each GUI was tested manually. We describe here the techniques.

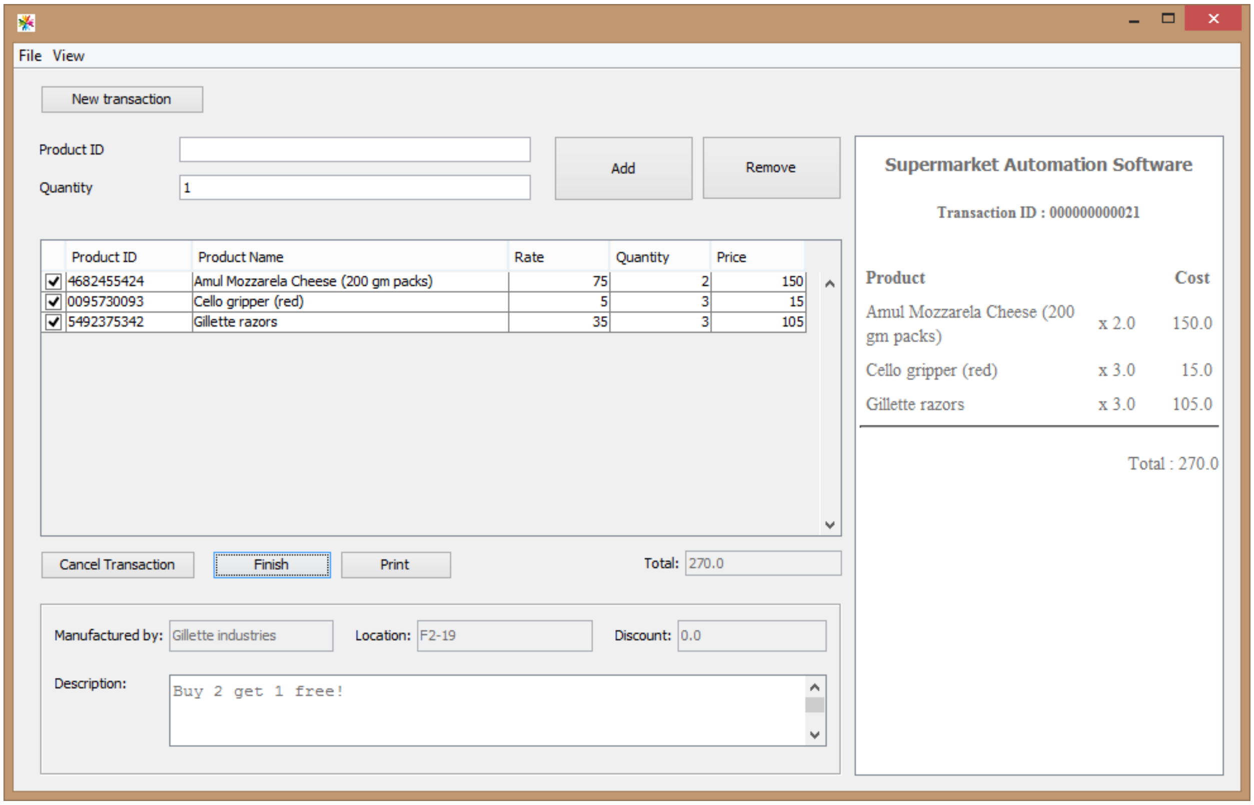
Login window

What is tested?	All those components which interact with the user.
Inputs	• User enters Employee ID and password • He presses login button
Expected result	If the details are correct, the main window will replace the login window; else an error message will be displayed in the login window.
Effective results	All results match the expected results.



Sales clerk main window

What is tested?	All those components which interact with the user.
Inputs	• Sales clerk enters product ID and quantity to be sold • He presses Add button • He sets the inclusion state for a product by checking/unchecking it. • He presses finish button. • He presses print button. • He presses cancel transaction button. • He presses logout menu item. • He presses Profile menu item. • He presses change password button.
Expected result	• The items list gets updated. If new item, it gets added to list. • The product is/is not added to the final list. • A print preview is generated. • The bill is printed for the transaction. • The current transaction is cancelled and a new transaction is started. • The current session is terminated and the window closes. • The profile widow opens showing the employee details. • The change password details window opens, where the password can be updated.
Effective results	All results match the expected results.



Product ID

Quantity 1

	Product ID	Product Name	Rate	Quantity	Price
☑	4682455424	Amul Mozzarela Cheese (200 gm packs)	75	2	150
☑	0095730093	Cello gripper (red)	5	3	15
☑	5492375342	Gillette razors	35	3	105

Total: 270.0

Manufactured by: Gillette industries

Location: F2-19

Discount: 0.0

Description: Buy 2 get 1 free!

Supermarket Automation Software

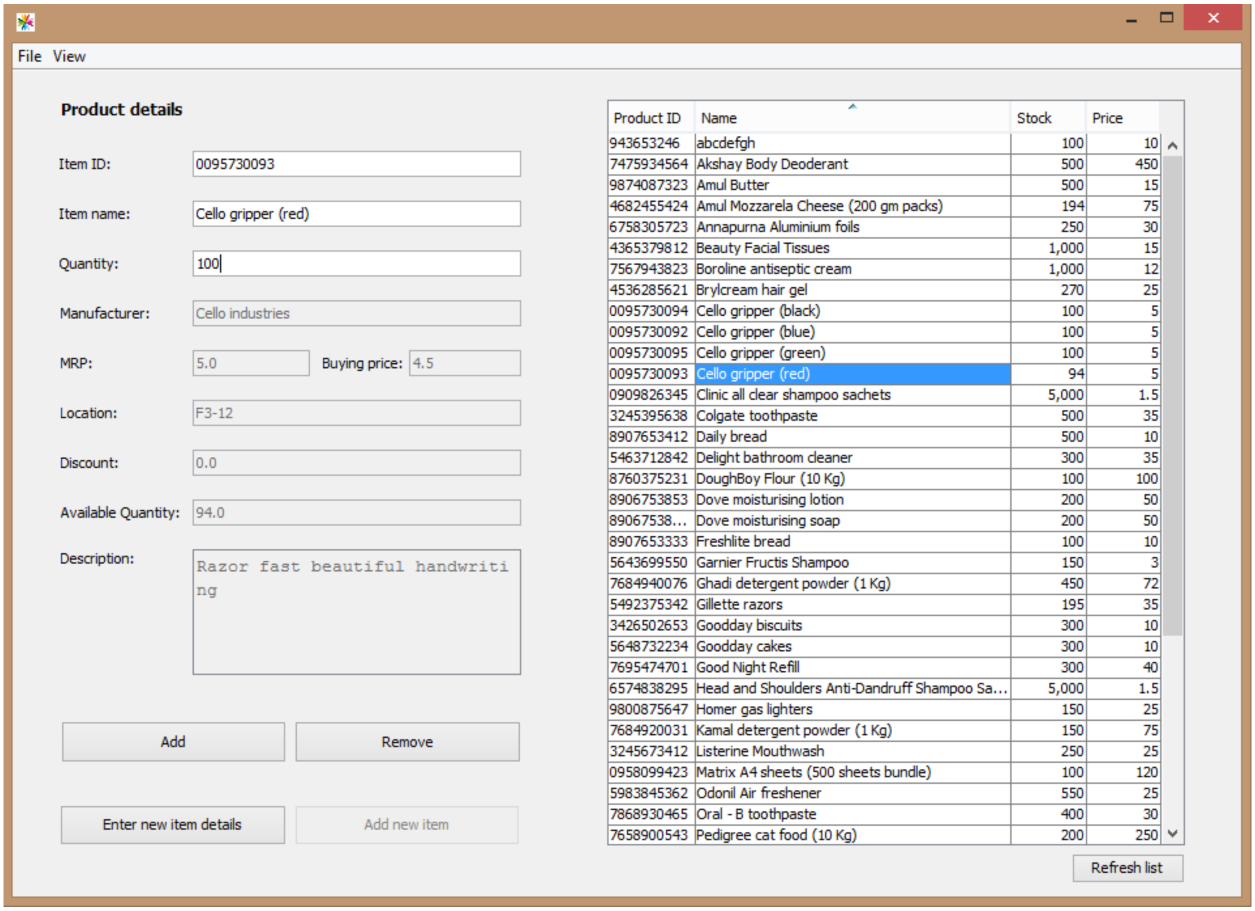
Transaction ID : 000000000021

Product		Cost
Amul Mozzarela Cheese (200 gm packs)	x 2.0	150.0
Cello gripper (red)	x 3.0	15.0
Gillette razors	x 3.0	105.0

Total : 270.0

Supermarket Staff main window

What is tested?	All those components which interact with the user.
Inputs	• Sales clerk enters product ID and quantity to be added/removed • He presses Add/Remove button. • He presses Add new item button. • He enters new item's details. • He presses save new item button. • He presses logout menu item. • He presses Profile menu item. • He presses change password button.
Expected result	• The items list gets updated accordingly. • Fields for setting product information turn editable. • Information for new information is stored and list is updated. • The current session is terminated and the window closes. • The profile widow opens showing the employee details. • The change password details window opens, where the password can be updated.
Effective results	All results match the expected results.



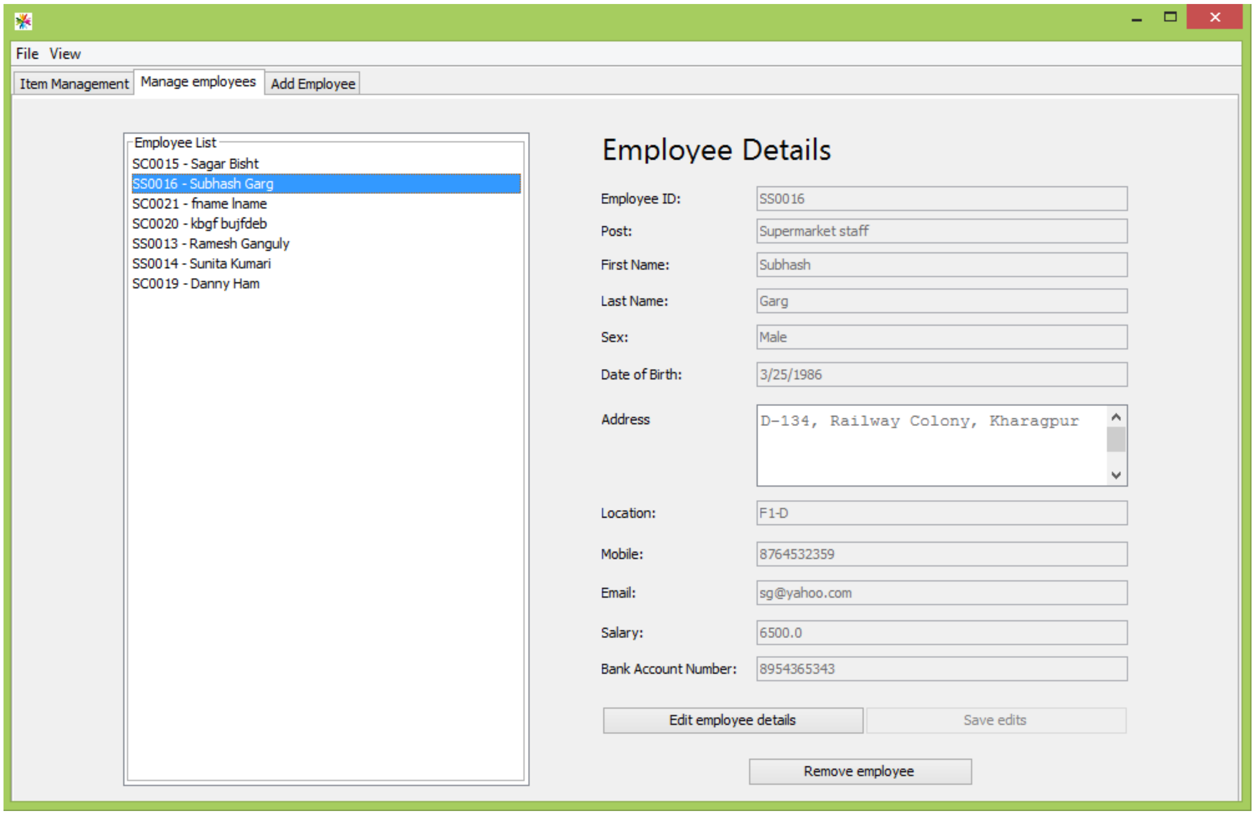
Product details

Item ID:	0095730093		
Item name:	Cello gripper (red)		
Quantity:	100		
Manufacturer:	Cello industries		
MRP:	5.0	Buying price:	4.5
Location:	F3-12		
Discount:	0.0		
Available Quantity:	94.0		
Description:	Razor fast beautiful handwriting		

Product ID	Name	Stock	Price
943653246	abcdefgh	100	10
7475934564	Akshay Body Deoderant	500	450
9874087323	Amul Butter	500	15
4682455424	Amul Mozzarela Cheese (200 gm packs)	194	75
6758305723	Annapurna Aluminium foils	250	30
4365379812	Beauty Facial Tissues	1,000	15
7567943823	Boroline antiseptic cream	1,000	12
4536285621	Brylcream hair gel	270	25
0095730094	Cello gripper (black)	100	5
0095730092	Cello gripper (blue)	100	5
0095730095	Cello gripper (green)	100	5
0095730093	Cello gripper (red)	94	5
0909826345	Clinic all clear shampoo sachets	5,000	1.5
3245395638	Colgate toothpaste	500	35
8907653412	Daily bread	500	10
5463712842	Delight bathroom cleaner	300	35
8760375231	DoughBoy Flour (10 Kg)	100	100
8906753853	Dove moisturising lotion	200	50
89067538...	Dove moisturising soap	200	50
8907653333	Freshlite bread	100	10
5643699550	Garnier Fructis Shampoo	150	3
7684940076	Ghadi detergent powder (1 Kg)	450	72
5492375342	Gillette razors	195	35
3426502653	Goodday biscuits	300	10
5648732234	Goodday cakes	300	10
7695474701	Good Night Refill	300	40
6574838295	Head and Shoulders Anti-Dandruff Shampoo Sa...	5,000	1.5
9800875647	Homer gas lighters	150	25
7684920031	Kamal detergent powder (1 Kg)	150	75
3245673412	Listerine Mouthwash	250	25
0958099423	Matrix A4 sheets (500 sheets bundle)	100	120
5983845362	Odonil Air freshener	550	25
7868930465	Oral - B toothpaste	400	30
7658900543	Pedigree cat food (10 Kg)	200	250

Manager window: Employee management

What is tested?	All those components which interact with the user.
Inputs	- Manager selects any employee from list by selecting him. - He presses edit button. - He enters new details for employee. - He presses save changes button. - He presses logout menu item. - He presses Profile menu item. - He presses change password button.
Expected result	- Details for the employee are set in the corresponding fields. - Information fields for employee become editable. - Information for the employee is updated. - The current session is terminated and the window closes. - The profile widow opens showing the employee details. - The change password details window opens, where the password can be updated.
Effective results	All results match the expected results.



Employee List

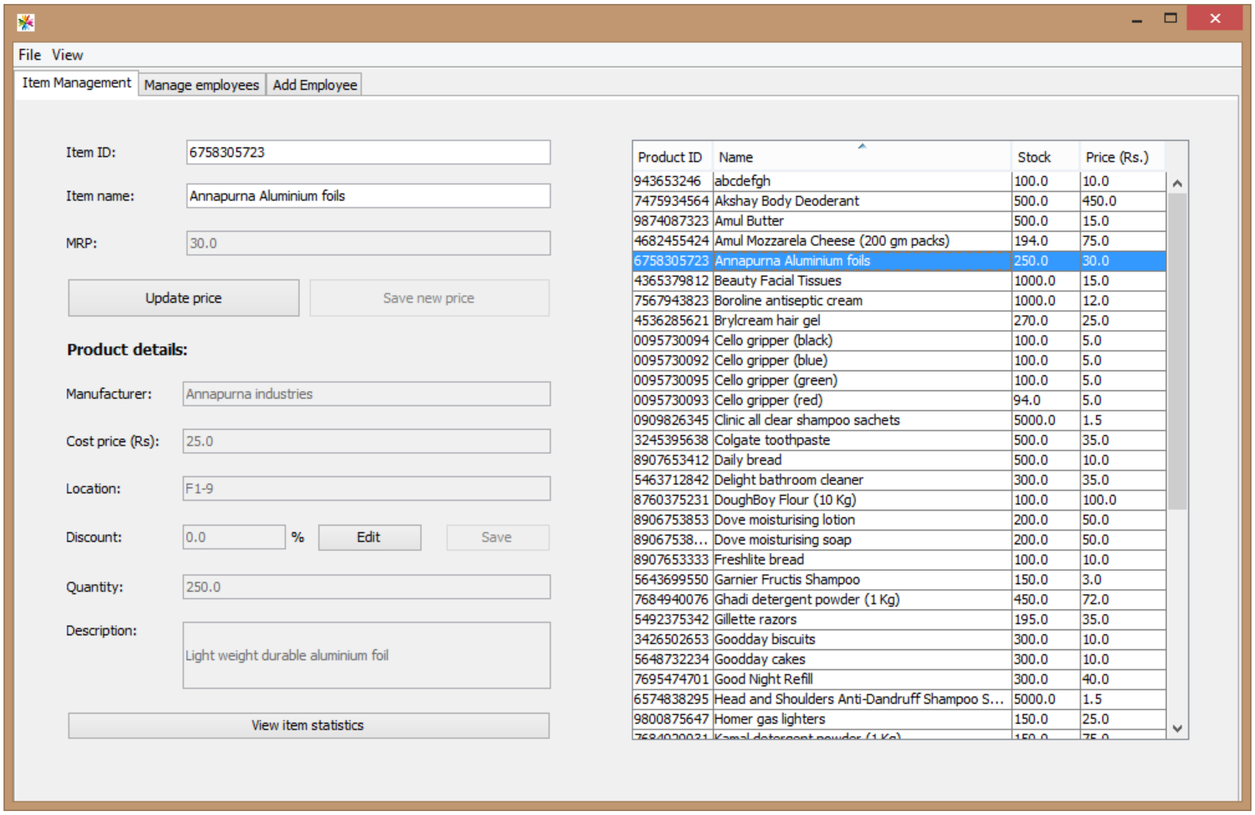
SC0015 - Sagar Bisht
SS0016 - Subhash Garg
SC0021 - fname lname
SC0020 - kbgf bujfdeb
SS0013 - Ramesh Ganguly
SS0014 - Sunita Kumari
SC0019 - Danny Ham

Employee Details

Employee ID:	SS0016
Post:	Supermarket staff
First Name:	Subhash
Last Name:	Garg
Sex:	Male
Date of Birth:	3/25/1986
Address	D-134, Railway Colony, Kharagpur
Location:	F1-D
Mobile:	8764532359
Email:	sg@yahoo.com
Salary:	6500.0
Bank Account Number:	8954365343

Manager window: Item management

What is tested?	All those components which interact with the user.
Inputs	- Manager selects any item from list by selecting it. - He presses edit price/edit discount button. - He enters new price/discount for product. - He presses save price/save discount changes button. - He presses logout menu item. - He presses Profile menu item. - He presses change password button.
Expected result	- Details for the product are set in the corresponding fields. - Price/discount fields for product become editable. - Information for the product is updated. - The current session is terminated and the window closes. - The profile widow opens showing the employee details. - The change password details window opens, where the password can be updated.
Effective results	All results match the expected results.



Item ID: 6758305723

Item name: Annapurna Aluminium foils

MRP: 30.0

Product details:

Manufacturer: Annapurna industries

Cost price (Rs): 25.0

Location: F1-9

Discount: 0.0 %

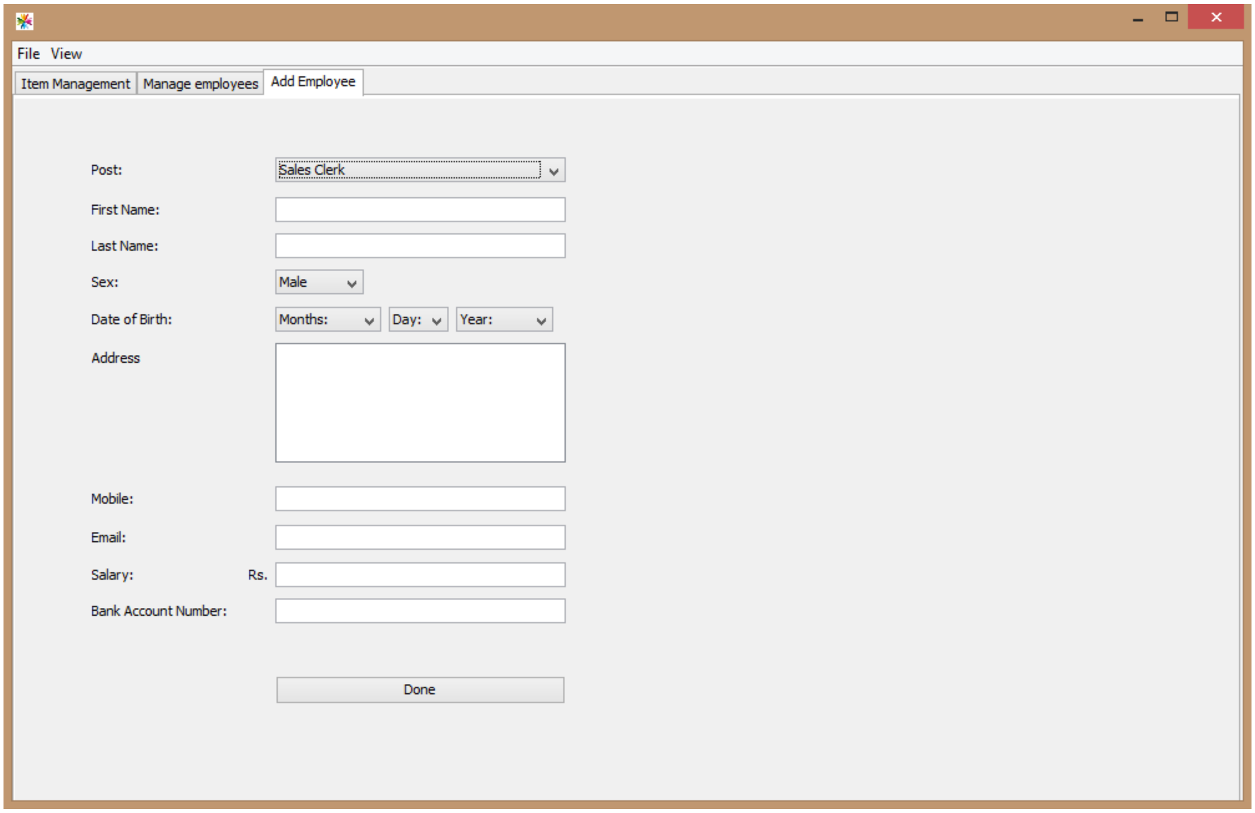
Quantity: 250.0

Description: Light weight durable aluminium foil

Product ID	Name	Stock	Price (Rs.)
943653246	abcdefgh	100.0	10.0
7475934564	Akshay Body Deoderant	500.0	450.0
9874087323	Amul Butter	500.0	15.0
4682455424	Amul Mozzarela Cheese (200 gm packs)	194.0	75.0
6758305723	Annapurna Aluminium foils	250.0	30.0
4365379812	Beauty Facial Tissues	1000.0	15.0
7567943823	Boroline antiseptic cream	1000.0	12.0
4536285621	Brylcream hair gel	270.0	25.0
0095730094	Cello gripper (black)	100.0	5.0
0095730092	Cello gripper (blue)	100.0	5.0
0095730095	Cello gripper (green)	100.0	5.0
0095730093	Cello gripper (red)	94.0	5.0
0909826345	Clinic all clear shampoo sachets	5000.0	1.5
3245395638	Colgate toothpaste	500.0	35.0
8907653412	Daily bread	500.0	10.0
5463712842	Delight bathroom cleaner	300.0	35.0
8760375231	DoughBoy Flour (10 Kg)	100.0	100.0
8906753853	Dove moisturising lotion	200.0	50.0
89067538...	Dove moisturising soap	200.0	50.0
8907653333	Freshlite bread	100.0	10.0
5643699550	Garnier Fructis Shampoo	150.0	3.0
7684940076	Ghadi detergent powder (1 Kg)	450.0	72.0
5492375342	Gillette razors	195.0	35.0
3426502653	Goodday biscuits	300.0	10.0
5648732234	Goodday cakes	300.0	10.0
7695474701	Good Night Refill	300.0	40.0
6574838295	Head and Shoulders Anti-Dandruff Shampoo S...	5000.0	1.5
9800875647	Homer gas lighters	150.0	25.0
7684920031	Kamal detergent powder (1 Kg)	150.0	75.0

Manager window: add employees

What is tested?	All those components which interact with the user.
Inputs	• Manager enters details for new employee. • He presses add employee button. • He presses logout menu item. • He presses Profile menu item. • He presses change password button.
Expected result	• New employee is added to the database. • The current session is terminated and the window closes. • The profile widow opens showing the employee details. • The change password details window opens, where the password can be updated.
Effective results	All results match the expected results.



Post: Sales Clerk

First Name:

Last Name:

Sex: Male

Date of Birth: Months: Day: Year:

Address

Mobile:

Email:

Salary: Rs.

Bank Account Number:

Entry and Exit Criteria

This section describes the general criteria by which testing commences, temporarily stopped, resumed and completed within each testing phase. Different features/components may have slight variation of their criteria, in which case, those should be mentioned in the feature test plan. The testing phase also maps to the impact level definition when a defect is entered in the bug-tracking phase.

Unit Testing

Unit Testing is done at the source or code level for language-specific programming errors such as bad syntax, logic errors, or to test particular functions or code modules. The unit test cases shall be designed to test the validity of the programs correctness.

Black Box Phase

Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. We will use Error Guessing and Boundary Value Analysis complexity metrics in order to quantifiably determine how many test cases needed to achieve maximum code coverage.

Black Box Entry Criteria

The Black Box Entry Criteria will rely on the component specification, and user interface requirements. Things that must be done on entry to the Black Box stage:

- All functions like sales transaction, employee management, viewing statistics, inventory management, etc. must either be coded or stubs written.
- The type of Black Box testing Methods will be determined upon entry. We will use Error Guessing, and Boundary Value Analysis.
- Error Guessing included entering garbage string in search field, trying to add the same product ID multiple times to the database, closing the server and starting application, entering invalid inputs for different fields.
- Boundary Value Analysis included adding a high number (like 1000) of items to the database, logging a large number of employees and updating inventory simultaneously.

Black Box Exit Criteria

The Black Box Exit Criteria listed below explains what needs to be completed in-order to exit Black Box phase. To exit the Black Box phase 100% success rate must be achieved. Things that must be done upon exiting the Black Box stage:

- The application showed no results in case of garbage string.
- If a product ID existing in the database was entered at the time of adding a new product, an error message shown.
- For a very high stress on the database, the response time of the server was increased.
- All code bugs that are exposed were corrected whenever possible.

White Box Phase

The White Box criteria apply for purposes of focusing on internal program structure, and discover all internal program errors. Defects were categorized and the quality of the software was assessed.

White Box Entry Criteria

The White Box Entry Criteria relied on verifying that the major features work separately but not necessarily in combination. The design and human interface were stable. Things that were done on entry to the White Box stage:

- Unit tests were written for as many functions as possible.
- The type of White Box testing Methods that were used were determined upon entry. We used unit testing and test for memory leaks.
- Black Box Testing was in its late stages.

White Box Exit Criteria

The Supermarket Automation Software in the White Box stage generally had a stable feel to it. White Box testing continued until the Black Box or next milestone criteria were met. To exit the White Box phase 100% success rate was achieved. The following describes the state of the product upon exit from the White Box Stage:

- All functions like Sales transaction, inventory management, employee management, viewing statistics were implemented, operational and tested.
- All test cases were generated. The test cases were generated from the Control Flow diagrams of all functions.
- The graphical interface was reviewed and found to satisfactory and stable, that is, no further changes to dialog boxes or other interface elements were planned. Minor changes were acceptable, but must be arranged with the Development and Test Engineers.
- All code bugs that were exposed were corrected.

Integration Test

There are two modules that will be integrated for Integration Testing. The two modules are The Graphic User Interface module and the Controller (back-end). The two components consists of a mixture of stubs, driver, and fully functional code. The following describes the entry and exit criteria for Integration testing.

Integration Test Entry Criteria

The Integration Test Entry Criteria relies on both modules to be operational. The controller and human interfaces were stable. Things that were done on entry to the Integration Test stage:

- All functions like Sales transaction, inventory management, employee management, viewing statistics were either be coded and/or stubs created.
- The Graphical User Interface was either be coded and/or a driver and stubs were created. The driver was implemented to facilitate test case input and output values.
- Interfaces and interactions between the Controller and the Graphical User Interface was operational.
- A bottom-up Integration Test Strategy was conducted. The low level details of controller and graphical interface were integrated. A driver was written to facilitate test case input and output values. The driver temporarily satisfied high-level details of the input and output values.

Integration Test Exit Criteria

The Integration Test Exit Criteria relied on both modules to be operational. The controller and human interface was stable. To exit the Integration Testing phase 100% success rate was achieved. Things that were done on exit from the Integration Test stage:

- All code bugs that were exposed were corrected.
- The parser and Graphical User Interface Module interacted together with complete accuracy, according to the System Specification Design. All discrepancies were corrected.
- Both Modules were ready for System Testing. Stubs and drivers were replaced with fully functional code.
- Black Box Testing was completed.

System Test

The System Test criteria apply for purposes of categorizing defects and the assessing the quality level of the product. All elements of the Controller and Graphical User Interface were meshed together and tested as a whole. System test focuses on functions and performance, reliability, instillation, behavior during special conditions, and stress testing.

1.3 Shipping or Live Release

The Controller and server testing was scaled down and combined all phases of testing into two phases – Function Complete and Regression testing – and follows the release criteria.

Shipping/Live Release Entry Criteria

The criteria for entering the final stages are as follows:

- All open product defects, regardless of fixed defects, documented, deferred, or otherwise addressed were identified.
- Regression testing on all product defects and the entire product was completed and verified.

Deliverables

- Program function specifications
- Program source code
- Test plan document - this document should address testing objectives, criteria, standards, schedule and assignments, and testing tools.
    - Unit Testing Plan
    - Integration Plan
    - System Testing Plan

Environmental Needs

As the project has been developed in Java, the software works in both Windows and Linux platforms.

MySQL is also required to function to be able to connect to the database.