EPOS SYSTEM TEST LOG

Graded Unit 2 – A Student

# Test Plan

**Functional Requirements Testing**

1. Log on using staff ID
2. Select product button
3. Display running total
4. Store payment as cash
5. Store payment as card
6. Clear sale
7. Refund sale
8. Add 5p bag
9. Produce end of day report
10. Print end of day report
11. Display stock list
12. Show change amount due
13. Deduct item from stock list when sold
14. Update stock list

**Additional Testing**

1. Remove button/product from stock list
2. Add new order
3. Edit existing product details
4. Print receipt

**Test Strategy**

The test strategy used will be white box. All functional requirements will be tested to ensure the application meets the specification requirements. White box testing has been chosen because the developer will be required to carry out testing. Additional testing will also be done on non-functional requirements that have been met. The following tests have been designed to check functionality and input validation.

**Unit Tests**

Each test has been designed for a specific unit of the application. This ensures each function meets the specification requirements. Some tests will incorporate multiple units, this is required to validate results.

**Usability testing**

Each test has been designed to check how easily the user can navigate the interface. The application has been designed using a simplistic concept, this helps reduce the time needed to perform functions.

**Tests**

**Log on**

Ensure user can log on with valid staff id

Ensure user cannot log on without valid user id

**Log off button**

Ensure main screen closes

Ensure log on form displays

**Select product button**

Ensure correct product names appears on sales screen

Ensure correct price for product appears on sales screen

**Display running total**

Ensure sales screen shows running total of order

Ensure sales screen correctly calculates sum of total product buttons entered

**Store payment as cash and save receipt**

Ensure user can place an order which is logged as cash

Ensure database logs sale as cash

Ensure receipt saves successfully

**Store payment as card**

Ensure user can place an order which is logged as card

Ensure database logs sale as card

**Clear sale**

Ensure user can clear sales screen by pressing the clear screen button

**Refund a sale**

Ensure user can search orders

Ensue user can select to refund searched order

Ensure deleted order no longer exists

**Add bag**

Ensure add bag button adds 5p to sale total

**Produce end of day report**

Ensure total card sales if shown

Ensure total cash sales if shown

Ensure total sales is shown

Ensure sales journal id increments by one when end of day button selected

**Print end of day report**

Ensure user can select to print end of day button

**Display stock list**

Ensure stock list is displayed to user

**Show change amount due**

Ensure change amounts screen opens when cash button selected

Ensure user can enter amount given by customer

Ensure correct change calculation is shown

**Deduct item from stock list when sold**

Ensure stock list has been deducted after order

**Update stock list**

Ensure user can add to stock list

Ensure stock list shows updated list

**Remove button/product from stock list**

Ensure user can remove an existing product from the Products table

**Add new order**

Ensure user can easily update stock quantities when a new order is received

**Edit existing product details**

Ensure user can edit price of product

Ensure user can edit quantity of product

Ensure user can edit description of product

# Test Cases

**Test Case 1**

**Log on**

* Ensure user can log on with valid staff id (101).
* Ensure user cannot log on without valid user id.

Functional requirement met: 1

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| 101 | Main screen shows, log on screen closes | Ok |
| 102 | Message box displaying “Invalid Staff ID, Please Re-Enter” | Ok |
| 2 | Message box displaying “Invalid Staff ID, Please Re-Enter” | Ok |
| 3 | Message box displaying “Invalid Staff ID, Please Re-Enter” | Ok |
| -4 | Message box displaying “Invalid Staff ID, Please Re-Enter” | Ok |
| dfsafs | Message box displaying “Please enter a valid staff ID” | Ok |
| Password | Message box displaying “Please enter a valid staff ID” | Ok |

**Test Case 2**

Log off button

1. Ensure main screen closes.
2. Ensure log on screen appears.

Functional requirement met: 1

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click Log Off button | Main screen closes, Log on screen opens | Ok |
| Double click Log Off button | Main screen closes, Log on screen opens | Ok |

**Test Case 3**

Select product button

1. Ensure correct product names appears on sales screen.
2. Ensure correct price for product appears on sales screen.

Functional requirement met: 2

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1) Click button with text “1. Gloves” | “Gloves £9.99” appears on sale screen | Ok |
| (1) Double Click button with text “1. Gloves” | “Gloves £9.99” appears on sale screen TWICE | Ok |
|  |  |  |
| (2) Click button with text “2. Hat” | “Hat £9.99” appears on sale screen | Ok |
| (2) Double click button with text “2. Hat” | “Hat £9.99” appears on sale screen TWICE | Ok |
|  |  |  |
| (3) Click button with text “3. T-Shirt” | “T-Shirt £29.99” appears on sale screen | Ok |
| (3) Double Click button with text “3. T-Shirt” | “T-Shirt £29.99” appears on sale screen TWICE | Ok |
|  |  |  |
| (4) Click button with text “7. Guard” | “Guard £19.99” appears on sale screen | Ok |
| (4) Double Click button with text “7. Guard” | “Guard £19.99” appears on sale screen TWICE | Ok |
|  |  |  |
| (5) Click button with text “18. new” | Message box displaying “Product out of stock appears” | Ok |

**Test Case 4**

Display running total

1. Ensure sales screen shows running total of order.
2. Ensure sales screen correctly calculates sum of total product buttons entered.

Functional requirement met: 3

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1) Click button with text “8. shorts” | Running total shows “Total £29.99” | Ok |
| (1) Click button with text “9. Cap” | Running total shows “Total £44.98” | Ok |
| (1) Click clear sale | Running total shows “Total £0.00” | Ok |
|  |  |  |
| (2) Click button with text “10. Bottle” | Running total shows “Total £9.99” | Ok |
| (2) Click button with text “12. Lights” | Running total shows “Total £59.98” | Ok |
| (2) Click clear sale | Running total shows “Total £0.00” | Ok |

**Test Case 5**

Store Payment as cash

1. Ensure user can place an order which is logged as cash
2. Ensure database logs sale as cash
3. Ensure receipt save successfully

Functional requirement met: 4, 12, 18

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click button with text “11. Helmet” | “Helmet £29.99” appears on sale screen | Ok |
| Click CASH button | Sales screen clears | Ok |
|  | Change Calculator screen shows | Ok |
| Enter 30 into “Enter amount given” text box |  |  |
| Click enter button on form | Change Amount text box shows “£0.01” | Ok |
| Click close button | Change Calculator form closes  Print dialogue box opens | Print dialogue box opens twice |
| Choose directory c:\user\documents | File dialogue box opens  File saves in c:\users\documents | Ok |
| Save file as receipt1.txt | Files saves as receipt1.txt | OK |
| Open receipt1.txt in c:\user\documents | File shows:  Order Number: (random)  Order Total £29.99  Date: (todays date & time)  Thank you for your business | Ok |
| Open Orders table | New row with following values   * OrderNo with random number up to 5 digits * Today’s date | Ok |
| Open Payments table | New row with following values   * OrderNo with random number up to 5 digits * PaymentAmount of £29.99 * PaymentType of “cash” | Ok |

**Test Case 6**

Store Payment as card

1. Ensure user can place an order which is logged as card
2. Ensure database logs sale as card

Functional requirement met: 5

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click button with text “5. Bike1” | “Bike1 £499.99” appears on sale screen | Ok |
| Click CARD button | Sales screen clears | Ok |
|  | Change Calculator screen shows | Ok |
| Enter 500 into “Enter amount given” text box |  |  |
| Click enter button on form | Change Amount text box shows “£0.01” | Ok |
| Click close button | Change Calculator form closes  Print dialogue box opens for receipt | Print dialogue box opens twice |
| Choose directory and name for receipt | File dialogue box opens | Ok |
| Open Orders table | New row with following values   * OrderNo with random number up to 5 digits * Today’s date | Ok |
| Open Payments table | New row with following values   * OrderNo with random number up to 5 digits * PaymentAmount of £499.99 * PaymentType of “card” | Ok |

**Test Case 7**

Clear sale

1. Ensure user can clear sales screen by pressing the clear screen button.

Functional requirement met: 6

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1) Click button with text “6. Bike2” | “Bike2 £599.99” appears on sales screen | Ok |
| (1) Click CLEAR SALE button | Sales screen clears | Ok |
|  |  |  |
| (2) Click button with text “13. Kids Bike” | “Kids Bike £49.99” appears on sales screen | Ok |
| (2) Click button with text “1. Gloves” | “Gloves £9.99” appears below “Kids Bike £49.99” on sales screen | Ok |
| (2) Click CLEAR SALE button | Sales screen clears | Ok |

**Test Case 8**

Refund sale

1. Ensure user can search orders
2. Ensure user can select to refund searched order
3. Ensure deleted order no longer exists

Functional requirement met: 7

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click REFUND button | Refund form opens | Ok |
| Enter “eerere” into Order Number text box |  |  |
| Click Search button | Message box opens displaying “Please enter a valid order number” | Ok |
| Enter 63464 in Order Number text box |  |  |
| Click Search Button | Data grid displays OrderNo: 63464 and OrderDate: 24/03/2017 | Ok |
| Click REFUND BUTTON | Order Number text box and data grid clears | Ok |
| Click CLOSE button | * Refund form closes * Order table no longer holds row identified with OrderNo 63464 * Payments table no longer holds row identified with OrderNo 63464 * SalesJournal table has a new entry with unique entry\_id, current journal\_id, -9.99 for card\_sales, 0 for cash\_sales and -9.99 for total\_sales. | Ok |

**Test Case 9**

Add 5p bag

1. Ensure add bag button adds 5p to sale total

Functional requirement met: 8

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click ADD BAG button | Sales screen show “Bag £0.05” | Ok |
| Double click ADD BAG button | Sales screen show “Bag £0.05” TWICE | Ok |

**Test Case 10**

Produce end of day report

1. Ensure total card sales if shown
2. Ensure total cash sales if shown
3. Ensure total sales is shown
4. Ensure sales journal id increments by one when end of day button selected

Functional requirement met: 9, 10

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (For reset before test) Click END OF DAY button | EndOfDay form opened | Ok |
| (For reset before test) Click Close button | EndOfDay form closes | Ok |
|  |  |  |
| Click button with text “1. Gloves” | “Gloves £9.99” appears on sales screen | Ok |
| Click button with text “2. Hat” | “Hat £9.99” appears on sales screen under “Gloves £9.99” | Ok |
| Click CASH button | ChangeCalculator form opens | Ok |
| Enter 20 in “Enter Amount Given” text box |  |  |
| Click enter button on form | Change Amount text box shows “£0.01” | Ok |
| Click close button | ChangeCalculator from closes | Ok |
|  |  |  |
| Click button with text “3. T-Shirt” | “T-Short £29.99” appears on sales screen | Ok |
| Click CARD button | ChangeCalculator form opens | Ok |
| Enter 40 in “Enter Amount Given” text box |  |  |
| Click enter button on form | Change Amount text box shows “£10.01” | Ok |
| Click close button | ChangeCalculator from closes | Ok |
| Click END OF DAY button | EndOfDay form opens  Values shown;   * Card Sales: 29.99 * Cash Sales: 19.98 * Total Sales: 49.97 | Ok |
| Click Print button | Print dialogue box opens | Print dialogue box opens twice. |
| Click Close button | EndOfDay form closes | Ok |

**Test Case 11**

Display stock list

1. Ensure stock list is displayed to user
2. Ensure stock list matches Products table

Functional requirement met: 11

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click View Stock List Button | StockListViewer form opens  Data grid shows complete Products table | Ok |
| Open Products table in database | Values compare successfully | Ok |

**Test Case 12**

Deduct item from stock list when sold

1. Ensure stock list has been updated

Functional requirement met: 13

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click View Stock List button | Opens StockListViewer form  Product with ProductID 5 has a quantity value of 5 | Ok |
| Click Close button | StockListViewer from closes | Ok |
| Click button with text “5. Bike1” | “Bike1 £499.99” appears on sale screen | Ok |
| Click CARD button | Sales screen clears | Ok |
|  | Change Calculator screen shows | Ok |
| Enter 500 into “Enter amount given” text box |  |  |
| Click enter button on form | Change Amount text box shows “£0.01” | Ok |
| Click close button | Change Calculator form closes | Ok |
| Click View Stock List button | Opens StockListViewer form   * Product with ProductID 5 has a quantity value of 4 (deducted after sale) | Ok |
| Click Close button | StockListViewer form closes | Ok |

**Test Case 13**

Update stock list

1. Ensure user can add to stock list
2. Ensure stock list shows updated list

Functional requirement met: 14

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1) Click “New Product Button” button | AddItem form opens  eposForm closes | Ok |
| (1) Enter 18 in Button Number text box | 18 is displayed in Button Number text box | Ok |
| (1) Enter 9.99 in Price text box | 9.99 is displayed in Price text box | Ok |
| (1) Enter 5 in Quantity text box | 5 is displayed in Quantity text box | Ok |
| (1) Enter Scarf in Description text box | Scarf is displayed in Description text box | Ok |
| (1) Click Add Product Button | Message box displaying “Updated appears”  AddItem form closes  eposForm opens  Button number 18 displays “18. Scarf” text | Ok |
| (1) Click button with text “18. Scarf” | Sales screen shows “Scarf £9.99” | Ok |
|  |  |  |
| (2) Click “New Product Button” button | AddItem form opens  eposForm closes | Ok |
| (2) Enter “eighteen” in Button Number text box | “eighteen” is displayed in Button Number text box | Ok |
| (2) Enter “nine” in Price text box | nine is displayed in Price text box | Ok |
| (2) Enter “five” in Quantity text box | five is displayed in Quantity text box | Ok |
| (2) Enter Hoodie in Description text box | Scarf is displayed in Description text box | Ok |
| (2) Click Add Product Button | Message box displaying “Please enter a valid product ID” | Ok |
| (2) Enter 19 in Button Number text box | 19 is displayed in Button Number text box | Ok |
| (2) Click Add Product Button | Message box displaying “Please enter a valid price” | Ok |
| (2) Enter 9.99 in Price text box | 9.99 is displayed in price text box | Ok |
| (2) Click Add Product Button | Message box displaying “Please enter a quantity” | Ok |
| (2) Enter 5 in Quantity text box | 5 is displayed in Quantity text box | Ok |
| (2) Click Add Product Button | Message box displaying “Updated appears”  AddItem form closes  eposForm opens  Button number 19 displays “19. Hoodie” text | Ok |

**Test Case 14**

Remove button/product from stock list

1. Ensure user can remove an existing product from the Products table

Functional requirement met: 15

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1) Click Remove Product Button | RemoveItem form opens  Main form closes | Ok |
| (1) Enter 13 in Button number | 13 appears in Button Number text box | Ok |
| (1) Click Remove Product Button | Confirmation form opens | Ok |
| (1) Click DELETE | Confirmation from closes | Ok |
| (1) Click Close | RemoveItem form closes  Main form opens with button 13 replaced with “new” | Ok |
| (1) Click button 13 | Product out of stock message appears | Ok |
|  |  |  |
| (2) Click Remove Button Product | RemoveItem form opens  Main form closes | Ok |
| (2) Enter 12 in Button number | 12 appears in Button Number text box | Ok |
| (2) Click Remove Product Button | Confirmation form opens | Ok |
| (2) Click Cancel | Confirmation from closes | Ok |
| (2) Click Close | RemoveItem form closes  Main form opens with no changes | Ok |
| (2) Click button 12 | “Lights £49.99” appears on screen | Ok |

**Test Case 15**

Add new order

1. Ensure user can easily update stock quantities when a new order is received

Functional requirement met: 16

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| Click View Stock List | StockListViewer form opens  Quantity for Product with ProductID 1 = 5 | Ok |
| Click Close | StockListViewer from closes | Ok |
| Click Add New Order | Add Order form opens | Ok |
| Enter 1 in Product ID text box | 1 appears in Product ID text box | Ok |
| Enter 5 in Quantity text box | 5 appears in quantity text box | Ok |
| Click Update | Message box with text “Updated” appears | Ok |
| Click ok in message box | Message box closes  StockListViewer from closes  Main form opens | Ok |
| Click View Stock List | StockListViewer form opens  Quantity for Product with ProductID 1 = 10 | Ok |
| Click Close | StockListViewer from closes | Ok |

**Test Case 16**

**Edit existing product details**

1. Ensure user can edit price of product
2. Ensure user can edit quantity of product
3. Ensure user can edit description of product

Functional requirement met: 17

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| (1)Click Edit Stock List | StockListEditor form opens  Main form closes | Ok |
| (1)Enter 14 in in Product ID text box | 14 appears in Product ID text box | Ok |
| (1)Enter 29.99 in Price text box | 29.99 appears in Price text box | Ok |
| (1)Enter 5 in Quantity text box | 5 appears in text box | Ok |
| (1)Enter “Red Jumper” in Description text box | “Red Jumper” appears in Description text box | Ok |
| (1)Click Update | Message box displaying “Updated appears” | Ok |
| (1)Click Ok on message box | StockListEditor form closes  Main form opens with newly updated butto/product 14 | Ok |
| (1)Click button 14 | “Red Jumper £29.99 appears” on sales screen | Ok |
|  |  |  |
| (2)Click Edit Stock List | StockListEditor form opens  Main form closes | Ok |
| (2)Enter “one” n Product ID text box | 14 appears in Product ID text box | Ok |
| (2)Enter “two” in Price text box | 29.99 appears in Price text box | Ok |
| (2)Enter “three” in Quantity text box | 5 appears in text box | Ok |
| (2)Enter “Green Jumper” in Description text box | “Red Jumper” appears in Description text box | Ok |
| (2)Click Update | Message box displaying “Please enter a valid Product ID” appears | Ok |
| (2)Click Ok in message box | Message box closes | Ok |
| (2)Enter 14 in Product ID text box | 14 appears in Product ID text box | Ok |
| (2)Click Update | Message box displaying “Please enter a valid Price” appears | Ok |
| (2)Click Ok in message box | Message box closes | Ok |
| (2)Enter 29.99 in Price text box | 29.99 appears in price text box | Ok |
| (2)Click Update | Message box displaying “Please enter a valid quantity” appears | Ok |
| (2)Click ok in message box | Message box closes | Ok |
| (2)Click Close | StockListEditor form closes  Main form opens with no changes | Ok |