A Testing Documentation Presented to the

Faculty of Datamex College of Saint Adeline, Inc

In Partial Fulfillment of the Requirements for the

Degree of Bachelor of Science in Information Technology

By:

Lorenzana,Joshua

**INTRODUCTION**

The **Kape Kalinaw Order Management System (OMS)** is a desktop program made with **VB.NET 2010.** It helps coffee shops manage their daily work more easily. The main purpose of the system is to make taking orders faster, reduce mistakes from manual recording, and give live updates on sales and orders. With this system, cafés can serve customers faster and keep accurate records for managing the business.

Small and medium coffee shops often have problems like wrong orders, trouble tracking money, calculation mistakes, and unclear sales reports. The Kape Kalinaw OMS solves these problems by giving a central and automatic system. It makes sure every order is recorded correctly, payments are calculated automatically, and all transactions are saved safely in the database. This saves staff time and helps management avoid mistakes with money.

The system has **two types of users: Admin** and **Cashier**. Admins have full control they can add, update, or delete products, change prices, see live sales information on the dashboard, and check past orders to correct mistakes. Cashiers focus on customer orders they can select products, choose the right size, add items to the order, process payments, and calculate change automatically. This separation of roles keeps work organized, protects important business information, and stops unauthorized actions.

The OMS also has a **dashboard and reports.** The dashboard shows live data like total orders, total sales, available products, and active users. It also shows detailed order history with order date, time, product name, size, quantity, and payment details. The system has charts that show sales trends over the last seven days. These tools help managers quickly understand business performance, find best-selling products, and plan improvements for the café.

**TESTING ENVIRONMENT**

Hardware Specifications:  
The system was tested on the following devices:

* Processor: Intel Core i5 or equivalent
* RAM: 8 GB
* Hard Drive: 500 GB or more
* Display: 1366×768 resolution or higher
* Operating System: Windows 10

Software Requirements:  
The testing was performed using:

* VB.NET 2010 (for running the OMS application)
* Microsoft SQL Server Express (for database)
* Windows Forms (for user interface)
* Microsoft Excel (optional, for reviewing test data or reports)

Test Data:  
Sample data was used to simulate real café operations. This included:

* Products like coffee, iced drinks, and non-coffee beverages with prices and images
* Customer orders with different sizes (16oz and 22oz)
* Payment amounts, including correct and incorrect entries
* Stock levels for cups, lids, and straws

This environment ensured the system could be tested in conditions similar to actual café usage.

**TESTING METHODOLOGY**

**Testing Approaches:**  
The Kape Kalinaw OMS was tested using the following approaches:

* **Black-box testing:** Checking the system’s functions without looking at the code. For example, verifying that orders are correctly added and payments are calculated.
* **White-box testing:** Checking the internal logic of the system, such as calculations for totals and stock deductions.
* **User Acceptance Testing (UAT):** Testing the system with actual users (Cashiers and Admins) to ensure it works as expected in a real café environment.

**Testing Tools:**

* **VB.NET 2010 IDE:** Used to run and debug the application.
* **SQL Server Express:** Used for database testing and verifying data storage.
* **Manual Testing:** Most tests were performed manually by entering sample data, placing orders, and checking reports.

**Test Cases and Criteria:**

* Test cases were designed to cover all main functions, including login, product selection, adding orders, processing payments, updating and deleting transactions, and generating reports.
* Each test case includes:
  + **Test Steps:** Actions taken to perform the test.
  + **Expected Output:** What should happen after the steps are completed.
  + **Actual Output:** What actually happened during testing.
  + **Status:** Pass or Fail based on whether the system worked correctly.
* The system passes testing if all main functions work correctly, stock levels update properly, and no critical errors occur.

**TEST CASES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Description | Test Steps | Expected Output | Actual Output | Status | Remarks |
| TC001 | Login with valid credentials | 1. Enter username 2. Enter password 3. Click login | User is redirected to dashboard | User is redirected to dashboard | Pass | N/A |
| TC002 | Login with invalid passwords | 1. Enter username 2. Enter incorrect password 3. Click login | Error message appears | No error message appeared | Fail | Bug identified |
| TC003 | |  | | --- | |  |  |  | | --- | | Add product to bill | | |  | | --- | |  |  |  | | --- | | 1. Select product 2. Choose size 3. Enter quantity 4. Click "Add to Billing" | | Product added to bill with correct total | Product added correctly | Pass | |  | | --- | |  |  |  | | --- | | N/A | |
| TC004 | |  | | --- | |  |  |  | | --- | | Process payment | | 1. Click "Process Transaction" 2. Enter valid payment amount | Payment processed, change displayed | Payment processed, change displayed | Pass | N/A |
| TC005 | Delete product from bill | 1. Select product in bill 2. Click "Delete Selected" | Product removed from bill, total updated | Product removed, total updated | Pass | |  | | --- | | N/A |  |  | | --- | |  | |
| TC006 | |  | | --- | |  |  |  | | --- | | Update product quantity/size | | 1. Select product 2. Change quantity or size 3. Click "Update Transaction" | Product details updated, total recalculated | Product details updated, total updated | Pass | N/A |
| TC007 | View orders report | 1. Go to dashboard 2. Check daily/weekly report | Orders report displays correct data | Orders report displays correct data | Pass | N/A |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bug ID | Description | Severity | Reported By | Status | Resolution |
| B001 | Login page crashes on incorrect password | High | Tester joshua | Open | fixed |
| B002 | "Add to Billing" button sometimes does not add the selected product | Medium | Tester joshua | |  | | --- | |  |  |  | | --- | | In Progress | | fixed |
| B003 | |  | | --- | |  |  |  | | --- | | Total price does not update correctly after deleting a product from bill | | High | Tester joshua | Open | fixed |
| B004 | Dashboard chart not updating in real-time | Medium | Tester joshua | |  | | --- | | In Progress |  |  | | --- | |  | | fixed |
| B005 | Search function does not filter products with special characters | |  | | --- | | Low |  |  | | --- | |  | | Tester joshua | Resolved | fixed |
| B006 | |  | | --- | |  |  |  | | --- | | Payment processing allows negative input | | Critical | Tester joshua | Resolved | fixed |

**BUG TRACKING & ISSUE LOG**

**USER ACCEPTANCE TESTING (UAT)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | |  | | --- | | **Scenario Description** |  |  | | --- | |  | | |  | | --- | | **Test Steps** |  |  | | --- | |  | | |  | | --- | | **Expected Result** |  |  | | --- | |  | | |  | | --- | | **Actual Result** |  |  | | --- | |  | | **Status** |
| UAT001 | |  | | --- | |  |  |  | | --- | | Login as Admin with valid credentials | | 1. Open OMS 2. Enter valid username and password 3. Click Login | Cashier is redirected to dashboard | Cashier successfully redirected | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT002 | |  | | --- | |  |  |  | | --- | | Login as Admin with valid credentials | | 1. Open OMS 2. Enter valid admin credentials 3. Click Login | Admin dashboard is displayed with full access | Admin dashboard displayed correctly | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT003 | |  | | --- | |  |  |  | | --- | |  |  |  | | --- | | Add product to bill | | 1. Select a product 2. Choose size and quantity 3. Click “Add to Billing” | Product is added to bill and total updated | Product added and total updated | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT004 | |  | | --- | |  |  |  | | --- | | Delete product from bill | | 1. Select product in bill 2. Click “Delete Selected” | |  | | --- | | Product removed and total recalculated |  |  | | --- | |  | | Product removed; total recalculated correctly | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT005 | Process transaction with insufficient payment | 1. Add products to bill 2. Click “Process Transaction” 3. Enter exact payment | Transaction completed and receipt displayed | |  | | --- | | Transaction successful |  |  | | --- | |  | | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT006 | |  | | --- | |  |  |  | | --- | | Process transaction with insufficient payment | | 1. Add products to bill 2. Click “Process Transaction” 3. Enter less than total amount | Error message displayed, transaction not processed | |  | | --- | | Error message displayed |  |  | | --- | |  | | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT007 | |  | | --- | |  |  |  | | --- | | Search for products | | |  | | --- | |  |  |  | | --- | | 1. Type product name in search box | | Only matching products are displayed | Search worked as expected | |  | | --- | | Pass |  |  | | --- | |  | |
| UAT008 | View sales dashboard | |  | | --- | | 1. Navigate to dashboard |  |  | | --- | |  | | Real-time data and charts displayed correctly | Dashboard charts updated | |  | | --- | | Pass |  |  | | --- | |  | |

**CONCLUSION & RECOMMENDATIONS**

The Kape Kalinaw Order Management System (OMS) successfully passed most of the planned test cases, including functional, user acceptance, and interface testing. Core features such as product selection, billing, transaction processing, role-based access, and reporting tools performed as expected. Minor issues, such as search functionality limitations and insufficient payment warnings, were identified and documented for future improvement.

Key Observations and Insights

* The system provides a user-friendly interface that allows both Cashiers and Admins to perform their tasks efficiently.
* Billing and transaction processing are accurate and automated, minimizing human errors.
* Role-based access control effectively restricts sensitive functionalities to Admin users.
* Dashboard and reporting features offer real-time insights, aiding management in decision-making.
* The few identified bugs are low to medium severity and do not compromise the core functionality.
* End-users provided positive feedback on the system’s intuitiveness and clarity.

Recommendations for Further Improvements

1. Enhance search functionality: Support partial matches, case-insensitive searches, and special characters for easier product lookup.
2. User feedback on actions: Include visual or sound cues for actions like deletion of items or insufficient payments.
3. Exporting reports: Add functionality to export sales reports to Excel or PDF formats for record keeping and analysis.
4. Mobile or web integration: Consider developing a web or mobile version for remote access and broader usability.
5. Performance optimization: Review database queries and UI rendering for faster loading times, especially with large datasets.

Overall, the OMS meets the intended objectives of streamlining order management, ensuring accurate transactions, and providing actionable business insights. Implementing the above recommendations will further enhance system usability and operational efficiency.