Warehouse Management System

Test Plan Document

CPSC-430: Writing Assignment 2
Dr. Anewalt
10 November 2017

Warehouse Group 2: Samantha Miller, Jacques Troussard, Taylor Dohmen

Table of Contents

Section 1: Introduction

1.1 Purpose

The purpose of this document is to describe and provide a detailed plan for testing the Warehouse Management System (WMS) project in Computer Science (CPSC)-430. The audience of this document includes the writers' professor Dr. Karen Anewalt and BGCVA's representative, Gusty Cooper.

1.2 Scope

The scope of this project to is provide a software application that manages automobile inventory for primary and mobile warehouses, including product management, listing, and invoice generation. Cooper represents the primary stakeholder in this project. The project will upload, display, and search products that BGCVA sells. This application intends to improve organization and track business statistics for the stakeholders.

1.3 Document Overview

This document is organized as follows:

- Section 2 provides a general description of the system and a list of requirements.
- Section 3 contains the test plan strategy, resources, products, and documentation.
- Section 4 lists the test procedure.
- Appendix A contains a glossary of commonly used terms.
- Appendix B provides examples and explanations of project documents.
- Appendix C enumerates the document contributions by each member in the group.

Section 2: Project Description

2.1 Project Scope

Warehouse Management System (WMS) will be an application by which BGCVA employees can manage inventory and sales records. It will digitally store information about each warehouse, the products in each warehouse, product sales, and the employees. Using WMS, employees should be able to query product and sales data, generate sales reports and invoices, and update inventory. Inventory updates will consist of operations related to transferring items between warehouses, adding new product deliveries into the system, and selling items to customers. All databases will be hosted on a server in the cloud, and will be accessible via a browser interface. Only employees will be able to access the system, and will need to login with a unique username and password to use the application.

2.2 Functional Requirements

2.2.1. Log into System
Description: A user accessing the site shall provide a email and password to use the application.
Main Flow:

1. User accesses site by Internet browser.
2. System displays login screen, prompting email and password fields, and a Login button.
3. User types email and password and clicks Login.
4. System verifies matching user information and displays the Home page of the site.

Alternate Flow A:

1. User accesses site page that requires login, like the Home or Import page.
2. The system redirects any user without a session to the Login page.

Alternate Flow B:

3. The user types either the email or password. The Login button is disabled until both fields are filled.

Alternate Flow C:

3. The user types a wrong email or password.
4. The system checks the provided information and returns a error message proclaiming the email or password was invalid.
5. Repeat until Main Flow 4 is met.

2.2.2. Log out of System
Description: A logged-in user shall exit their logged-in session on the site.
Main flow:

1. A logged-in user clicks the Logout button on the menu.

2. The system removes the user from the session and displays the Login page of the application.

2.2.3. Search for Product

Description: Any user shall search for products based on warehouse location, product name, or product number.

Main Flow:

1. A logged-in user navigates to the Search page via the menu bar.
2. The system displays the Search page, which offers options to search for products by warehouse location, product name, or product number, as well as Search and Reset buttons.
3. The user enters requirements in the form and clicks Search.
4. The system queries all related information and displays all product information in the table.

Alternate Flow A:

3. The user clicks the Reset button instead of Search.
4. The system clears the fields of the search boxes.

Alternate Flow B:

5. The user clicks the Reset button after searching.
6. The system clears the returned table and search fields.

2.2.4. Import Inventory File

Description: Any user shall upload a Comma Separated Values (CSV) file into the system which specifies what changes are being made to the warehouse inventory. The system shall then update the product database to account for the specified changes.

Main Flow:

1. User navigates to an inventory management page and selects the option to upload an inventory management file. (See Appendix B.1 for preliminary import configuration.)
2. System displays an Upload window including a File Explorer and Upload button.
3. User navigates the File Explorer, selects the desired file, and clicks the Upload button.
4. The system parses the file and updates the inventory database based on the file's specification. If the operation is time intensive, the system should display a progress indicator to the user.
5. The system displays a message informing the user that the operation was executed successfully.

Alternate Flow A:

4. The system parses the file and finds that it is improperly formatted.
5. The system displays a message informing the user that the file is improperly formatted. Return to Main Flow 1.

Alternate Flow B:

6. The system parses the file and finds that there is an invalid operation.
7. The System displays a message informing the user of bad operation. Return to Main Flow 1.

2.2.5. Create Invoice

Description: A sales associate shall select a subset of their mobile warehouse's inventory and create and invoice for the sale of those items.

Main Flow:

1. Sales associate navigates to a Create Invoice page.
2. System displays form with a field for entering customer information, a list of the products in the sales associate's mobile warehouse with areas to specify the sold quantity of each item, and a submit button.
3. Sales associate enters the customer's information, specifies the quantity of each item being sold, and selects submit.
4. System validates that the quantities are not larger than the quantity of the item in stock.
5. System generates invoice based on the price and quantities of the items sold and customer information given and saves the sale record in the database.
6. System updates inventory database to account for the sold items.
7. System displays message indicating that the sale was successfully documented.

Alternate Flow A:

4. System finds that a sale quantity is larger than the inventory quantity and displays a message indicating this error. Return to Main Flow 2.

2.2.6. Search for Invoice (by Sales Associate, Customer, Date)

Description: Any user should search for and review records of previous sales.

Main Flow:

1. User navigates to Sales page.
2. System displays a list of previous sales, sorted by date (most recent first), a search field, a drop-down menu to select search field, and a search button .
3. User selects what field they want to search by, enters a term into the search field, and selects search.
4. System queries the sales database to and displays a list of sales records that match the search terms.
5. User selects a particular search result.
6. System displays invoice and an option to download the invoice as a txt file for that particular sale.

2.2.7. Generate Sales Metrics (by associate between range of dates)

Description: A manager should generate a report detailing what each sales associate sold, given a date range.

Main Flow:

1. Manager navigates to Sales Metrics page.
2. System displays input fields for start date and end date and a generate report button.
3. Manager fills in the date fields and selects the Generate Report button.
4. System queries the sales database and compiles a report of metrics breakdown including total sales and sales per item each associate sold during the time period.

5. System displays a list of associates, their totals, and a sublist of items the quantity of each that the associate sold.

2.2.8. Create a User Account

Description: The administrator shall create any user accounts and the manager should create sale associate accounts, allowing unique sales associates to access their inventory stores.

Main Flow (Admin):

1. Administrator log into the postgresql database as an administrative user.
2. Administrator creates an inventory manager role with a preselected list of privileges allowing them to access and modify the 'users' table as well as all other tables necessary to facilitate the manager's functions.

Main Flow (Manager):

1. Manager logs into their account.
2. System compares credentials to SQL users table.
3. Manager navigates to manage users management page.
4. System queries users database and renders a table with users and check boxes
5. Manager selects 'Create User.'
6. System renders web form containing all the necessary fields required to insert a new user.
7. Manager finishes form and selects submit (reset simply clears form).
8. System receives post request from manager user, scrubs input and updates SQL users table.
9. System generates password and renders success page with username and password.

Alternate Flow (Manager):

9. System generates non-generic error and renders error page.

2.2.9. Update a User Account

Description: The manager should update the various attributes associated with any a user account of lesser privilege level.

1. Manager logs into their account.
2. System compares credentials to SQL user tables.
3. Manager navigates to user page.
4. System queries users database and renders a table with users and check boxes.
5. Manager selects user checkbox and clicks 'Update.'
6. System queries database for user and attributes.
7. System renders a web form preloaded with user attributes.
8. System logs date, time, manager id, and user's details.
9. Manager refills desired user attribute fields and selects 'Submit.'
10. System receives post request from manager user, scrubs input and updates SQL users table and appends changes to log.
11. System generates success page.

Alternate Flow:

10. System generates non-generic error and renders error page

2.2.10. Disable a User Account

Description: The manager should modify the access status to the system of any user account of lesser privilege level.

1. Manager logs into their account
2. System compares credentials to SQL user tables.
3. Manager navigates to user page.
4. System queries users database and and renders a table with users and check boxes.
5. Manager selects user checkbox and clicks ‘Update.’
6. System queries database for user and attributes.
7. System renders a web form preloaded with user attributes.
8. System logs date, time, manager id, and user’s details.
9. Manager toggles Disable/Enable radio button and selects submit.
10. System receives post request from manager user, scrubs input and updates SQL users table and appends changes to log.
11. System generates success page.

Alternate Flow:

10. System generates non-generic error and renders error page.

2.2.11. Update Product Attributes

Description: Inventory manage should access list of products queried from database and modify attributes individually.

Main Flow:

1. Manager logs into their account.
2. System compares credentials to SQL user tables.
3. Manager navigates to inventory management page.
4. System renders management page with link to modify item.
5. Manager click modify item link.
6. System renders page with part description field and submit button.
7. Manager enters item key terms (part number, name, etc.)
8. System queries database and renders table with possible matches.
9. Manager selects item and clicks on ‘Modify.’
10. System renders item page, a preloaded web form with item attributes.
11. Manager modifies data and clicks submit.
12. System receives post request from manager user, scrubs input and updates SQL users table and appends changes to log.

Alternate Flow:

12. System generates non-generic error and renders error page.

Section 3: Test Plan

3.1 Testing Strategy

The testing strategy relies on unit, integration, and usability testing. The site's developers shall provide their own unit testing, reviewing Python methods for lacking coverage for all functional and committed code to the master branch of the repository. These unit tests are not documented to maintain the flexibility needed for changes. The test group shall analyze the site for black-box testing: integration and usability testing. Integration testing refers to verifying components of the system work cohesively as a singular unit. Using itemized testing measures in Section 4, the test group will successfully engage with the site and document any issues that arise. Usability testing refers to the testing team providing feedback on the user interface based on the requirements provided in Section 2.

3.2 Testing Resources

The test suite for this project will be performed on the current deployment of the application in its production environment. It can be accessed via the url http://35.196.102.118. The application will be populated with an assortment of test data. In Appendix B.1 there is an example CSV file that can be used to test the file import functionality of the system. This document will work with the pre-populated test data, but since there are many ways of composing import files, the testing team should feel free to create one or more additional files that they think should work with the given test data and use them in testing. This is not strictly necessary, but if the test team is feeling particularly ambitious it will give us greater confidence in the reliability of the system.

3.3 Test Work Products

The test cases are enumerated in Section 4, and the testing team will be responsible for documenting the results while executing the suite. The developers recommend copying the included tables into an Excel sheet. Then as the tests are completed, testers shall fill out the blank columns with appropriate information describing results. For section 4.1, Integration Testing, the testing team should add two columns when they create their results spreadsheet: Pass/Fail and Description. The Pass/Fail column will be used to indicate whether the acceptance criteria was met for a given test, and if not the Description column should be used to denote what results were produced and exactly what parts of the criteria were not met. In the case of failure, it may be useful to include screenshots of any errors or unexpected behavior that occurs, but testing team may decide what information is appropriate on a case-by-case basis.

3.4 Test Documentation

The record keeping mechanism for this test plan is fairly straightforward. A simple form which lists the test number and purpose followed by four open columns which allow the tester to record their data. The following inputs from the user are required. Tester's input data, the actual results and if the action was successful or not. In addition to these main tests the system needs to be primed. The form will be prepended with these subsection numbers. Testers will need to check and confirm the priming of the system was completed. Additional forms that should be included with this form;

1. Custom import file created by testing team - See test 23 and 24
2. Generated invoice - see test 30

3.4.1 Priming

DO THE FOLLOWING BEFORE TRYING TO FILL OUT RECORD FORM

3.4.1.1 Ensure there is an active Admin account (implementation team duty)
3.4.1.2 Create a manager account.
3.4.1.3 Create 2 active associate accounts
3.4.1.4 Create 1 deactivated associate account

3.4.2 Examples

3.4.2.1 Primer Log

Section	Complete?
3.4.1	Y

3.4.2.2 Main Log

Test #	**Purpose**	**Your Input Data**	**Expected Results**	**Actual Results**	**Success**
1	Login as admin.	Login: jsmith Password: pass	Main page is rendered with personalized greeting.	Main page was rendered with personalized greeting "Welcome jsmith! Use	Y

				the menu bar above to get started. “	

3.4.3 Record Forms

Section	**Complete?**
3.4.1	
3.4.2	
3.4.3	
3.4.4	

Test #	**Purpose**	**Your Input Data**	**Expected Results**	**Actual Results**	**Success**
1	Login as admin.	Login: Password:	Main page is rendered with personalized greeting.		
2	Login as manager.	Login: Password:	Main page is rendered with personalized greeting.		
3	Login as associate.	Login: Password:	Main page is rendered with personalized greeting.		
4	Login as deactivated.	Email: Password:	User cannot log in and an explanation is provided.		
5	Logout as sales.	NA	The login page appears.		
6	Display page.	NA	Three searchable fields and no		

			results are displayed until searched.		
7	Return product search results.	product name: Product id number:	Found results are listed in a table. Results not found is handled correctly.		
8	Display page.	NA	Observe Admin page with Accounts link.		
9	Display page.	NA	Observe the list of system users and related information.		
10	Create (good) user.	User form input:	Created user is displayed on the Accounts page.		
11	Create (bad) user.	User form input:	Acceptable error is displayed on the Accounts page.		
12	Create user (cancelled).	NA	Transaction is not completed and user is returned to the Accounts page.		
13	Update (good) user.	Update form input:	Updated user is displayed on the Accounts page.		
14	Update (bad) user.	Update form input:	Acceptable error is displayed on the Accounts page.		

15	Update user (cancelled).	NA	Transaction is not completed and user is returned to the Accounts page.		
16	Disable user.	User id:	Observe the user account is disabled on the Accounts page.		
17	Display Page.	NA	List of all customers in alphabetical order by name. All should have update buttons.		
18	Create new customer.	Customer form input:	Return to the customer display page with an indicator of success or failure. If success the new customers info should be in the list.		
19	Update customer informatio n.	Update form input:	Return to the customer display page and see that the customer's information has been updated.		
20	Import page.	NA	Displays import page which includes a file upload form and example of each type of possible transaction that can be in a file.		
21	Select File	NA	A file selector		

			window. The file name should appear on the input page after selection.		
22	Cancel Selection	NA	The file name should disappear.		
23	Upload file that performs some warehouse transactions successfully.	Custom import file: (attach to form) Example import file provided:	On selecting upload the system should give a message indicating success.		
24	Upload file that has errors and fails system validation.	Custom import file: (attach to form)	On selecting browse you should be able to upload a local file to the system. After selecting upload the system should give a message indicating failure and enumerate the file errors.		
25	Render create invoice page.	NA	Should render a web form with the following fields; Customer ID, Product Part Number, and Qty.		
26	Add line item to invoice form.	Product id: Product number: Quantity:	On Selecting the ADD ANOTHER LINE button,		

	(Success)		another two fields should appear below the original product part number and quantity fields.		
27	Add line item to invoice form. (Error)	NA (Do not fill out product id field)	On selecting the ADD ANOTHER LINE button, a javascript alert informs the user that their previous line item's product id is still empty.		
28	Add line item to invoice form. (Error)	Product Id:	On selecting the ADD ANOTHER LINE button, a javascript alert informs the user that their previous line item's quantity is still empty.		
29	Create a hard copy .txt file of the actual invoice. (Success)	Invoice form inputs:	On selecting SUBMIT you should receive an alert indicating a successful submission.		
29	Create a hard copy .txt file of the actual invoice. (Error: incomplet e form)	Leave any of the form fields empty. Invoice form inputs:	On selecting SUBMIT you should be prompted to fill out the incomplete fields. The missing fields should also be		

			highlighted.		
30	Access hard copy of invoice.	Attach file to test plan document deliverables.	Should behave as attachment and prompt user to download the invoice as a .txt file.		
31	Create a hard copy .txt file of the actual invoice. (Error: Customer ID)	Customer id:	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.		
32	Create a hard copy .txt file of the actual invoice. (Error: Seller)	Seller id:	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.		
33	Create a hard copy .txt file of the actual invoice. (Error: Product Part Number)	Product id:	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional		

			information to help identify the error more specifically.		
34	Create a hard copy .txt file of the actual invoice. (Error: Quantity)	Seller: Product id: Qty:	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.		
35	Search Invoice Page	NA	Should display a keyword input field and a date range input field and no results.		
36	Invoice Search	Search term inputs:	A list of sales sorted by date that are within the given date range and are related in some way to the keyword search term.		

3.5 Test Schedule

This section contains all the data related to the test schedule. Test section descriptions will be organized in a table grouping the tests with their estimated time schedule as well as identifying the responsible testing party.

Milestone/Test	**Team**	**Start**	**End**	
Production Environment Test	Implementation	10/13/17	10/14/17	
Application Priming (Admin account creation)	Implementation	10/13/17	10/14/17	
Access tests	Implementation	10/14/17	10/15/17	
Application Priming (General account creation and editing)	Testing	10/15/17	10/17/17	3.4.1.x sub-sections
Access/Session Testing	Testing	10/15/17	10/17/17	Login and general access
DB modification via web application interface	Testing	10/15/17	10/17/17	Products, Customers, Accounts
DB modification via import file	Testing	10/15/17	10/17/17	
Invoicing	Testing	10/15/17	10/17/17	
Deliverables	Testing	10/15/17	10/17/17	Final forms

Section 4. Test Procedure

4.1 Integration Tests

Test #	Rqmt	Subsystem	Purpose	Input Data	Expected Results
1	2.2.1	Login	Login as admin.	Email: lc@test.com Password: 123	User logs in and accesses all pages.
2	2.2.1	Login	Login as manager.	Email: md@test.com Password: 123	User logs in and accesses all pages.
3	2.2.1	Login	Login as sales.	Email: jj@test.com Password: 123	User logs in and accesses all pages except Admin and Accounts pages.
4	2.2.1	Login	Login as deactivated.	Email: dr@test.com Password: 123	User cannot log in and an explanation is provided.
5	2.2.2	Logout	Logout as sales.	Click Logout on the header.	The login page appears.
6	2.2.3	Products	Display page.	Click the display page.	Three searchable fields and no results are displayed until searched.
7	2.2.3	Products	Return product search results.	Use the product name (BG CF5), number (PN 203), and warehouse id (0 to 5) to search for a variety of results.	Found results are listed in a table. Results not found is handled correctly.
8	2.2.8	Admin	Display page.	Click Admin on the menu bar.	Observe Admin page with Accounts link.
9	2.2.8	Accounts	Display page.	Click Accounts on the Admin page.	Observe the list of system users and related information.
10	2.2.8	Accounts	Create	Create user with a	Created user is

			(good) user.	variety of inputs (warehouse id is optional).	displayed on the Accounts page.
11	2.2.8	Accounts	Create (bad) user.	Create user with duplicate or missing information.	Acceptable error is displayed on the Accounts page.
12	2.2.8	Accounts	Create user (cancelled).	Create user but click Cancel.	Transaction is not completed and user is returned to the Accounts page.
13	2.2.9	Accounts	Update (good) user.	Update user with a variety of inputs (password and warehouse id is optional). Email cannot be changed.	Updated user is displayed on the Accounts page.
14	2.2.9	Accounts	Update (bad) user.	Update user with duplicate or missing information.	Acceptable error is displayed on the Accounts page.
15	2.2.9	Accounts	Update user (cancelled).	Update user but click Cancel.	Transaction is not completed and user is returned to the Accounts page.
16	2.2.10	Accounts	Disable user.	Update a user, changing role to “Disabled”. Click Update.	Observe the user account is disabled on the Accounts page.
17	2.2.5	Customers	Display Page.	Click Customers in the nav bar.	List of all customers in alphabetical order by name. All should have update buttons.
18	2.2.5	Customers	Create new customer.	Select new customer on customer display page. Fill out form with appropriate values.	Return to the customer display page with an indicator of success or failure. If success the new customers info should be in the list.

19	2.2.5	Customers	Update customer information.	Select update on one customer from the display page. Make some changes to the form. Select Submit.	Return to the customer display page and see that the customer's information has been updated.
20	2.2.4	File Import	Import page.	Select Import in the nav bar.	Displays import page which includes a file upload form and example of each type of possible transaction that can be in a file.
21	2.2.4	File Import	Select File	Select the browse button.	A file selector window. The file name should appear on the input page after selection.
22	2.2.4	File Import	Cancel Selection	After selecting a file, click the cancel button.	The file name should disappear.
23	2.2.4	File Import	Upload file that performs some warehouse transactions successfully.	The example file included in appendix B1 or a custom file made by the testing team.	On selecting upload the system should give a message indicating success.
24	2.2.4	File Import	Upload file that has errors and fails system validation.	The example file included in appendix B1 run a second time or some custom file made by the testing team.	On selecting browse you should be able to upload a local file to the system. After selecting upload the system should give a message indicating failure and enumerate the file errors.
25	2.2.5	Invoice Metrics	Render create invoice page.	Select invoice in the navigation bar, then select CREATE button.	Should render a web form with the following fields; Customer ID, Product Part

					Number, and Qty.
26	2.2.5	Invoice creation	Add line item to invoice form. (Success)	Fill out both product di number and quantity fields then select ADD ANOTHER LINE.	On Selecting the ADD ANOTHER LINE button, another two fields should appear below the original product part number and quantity fields.
27	2.2.5	Invoice creation	Add line item to invoice form. (Error)	Do not fill out product number id field then select ADD ANOTHER LINE.	On selecting the ADD ANOTHER LINE button, a javascript alert informs the user that their previous line item's product id is still empty.
28	2.2.5	Invoice creation	Add line item to invoice form. (Error)	Fill out product number id field, but not the quantity field then select ADD ANOTHER LINE.	On selecting the ADD ANOTHER LINE button, a javascript alert informs the user that their previous line item's quantity is still empty.
29	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Success)	The create invoice page requires all four fields to be filled out to attempt invoice creation. Customer ID, Seller Email, Product Part Number, and Quantity Sold.	On selecting SUBMIT you should receive an alert indicating a successful submission.
29	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Error: incomplete form)	Leave any of the form fields empty.	On selecting SUBMIT you should be prompted to fill out the incomplete fields. The missing fields should also be highlighted.
30	2.2.5	Invoice	Access hard	Upon successful	Should behave as

		creation	copy of invoice.	creation of invoice, follow link provided within alert.	attachment and prompt user to download the invoice as a .txt file.
31	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Error: Customer ID)	Enter a non existing customer id.	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.
32	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Error: Seller)	Enter a non existing seller email.	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.
33	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Error: Product Part Number)	Enter a non existing product part number.	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error more specifically.
34	2.2.5	Invoice creation	Create a hard copy .txt file of the actual invoice. (Error: Quantity)	Enter a quantity greater than what is available in the corresponding warehouse.	On selecting SUBMIT you should receive an alert indicating a there was an error followed by some additional information to help identify the error

					more specifically.
35	2.2.7	Invoice Metrics	Search Invoice Page	Select Invoice in the nav bar. Then select invoice search.	Should display a keyword input field and a date range input field and no results.
36	2.2.7	Invoice Metrics	Invoice Search	Any search term and/or date range. This should handle any possible combination including empty fields.	A list of sales sorted by date that are within the given date range and are related in some way to the keyword search term.

4.2 Usability Tests

Note: Usability field: 0 being low (not usable at all) to 5 (very easy to use and understand its purpose).

Test #	**Rqmt**	**Subsystem**	**Purpose**	**Usability (0-5)**	**Feedback**
1	2.2.1	Sessions	Login		
2	2.2.2	Sessions	Logout		
3	2.2.3	Products	Search for Product		
4	2.2.4	Import	Import Inventory File		
5	2.2.5	Invoice, Customers	Create Invoice		
6	2.2.6	Invoice	Search for Invoice		
7	2.2.7	Invoice	Sales Metric		
8	2.2.8	Accounts	Create User		

9	2.2.9	Accounts	Update User		
10	2.2.10	Accounts	Disable User		

Appendix A. Glossary

This appendix contains a list of abbreviations used in this document.

CPSC	Computer Science
CSV	Comma Separated Values
WMS	Warehouse Management System

Appendix B. System Documents

B.1 Product Import

The below text is an example of an import file format. # symbols indicate commented lines that will not be executed or validated. This file should work the first time it is imported. After that it will produce errors as it will have made changes to the inventory that render it invalid.

```
# new products
# toId, prodName, prodDesc, quant, price, serialNo
1, oil, oil for the engine, 100, 5.99, 1
1, wipers, a pair of windshield wipers, 100, 105.99, 2
1, headlight fluid, to power the headlights, 100, 54.99, 3
1, timing belt, to do important things inside vehicles, 100, 0.99, 4
1, radio, for da music, 100, 5555.99, 5

# transfers
# fromId, toId, prodId, quant
1, 3, 7, 1
1, 4, 10, 50
1, 2, 6, 100

# restocks
# toId, prodId, quant
1, 6, 100
2, 7, 20
```

B.2 Invoice Example

Below captures the text file created from the system when a sale is made.

Business Name
123 Example Dr.
Ashland, VA 23005
<phone number>
<email>

To:
Business Name
123 Example Dr.
Ashland, VA 23005
Cc. Manager Name

<BG Distributive Group Company Name> Invoice

Invoice Number: ##
Date: mm-dd-yyyy

Product Name	Qty	Price	Product Number	Total Amount
Test 1	1	$56.00	10100	$56.00
Test 2	3	$15.00	16534	$45.00

Total $101

Appendix C. Document Contributions

This appendix lists each group member's document contributions.

- Samantha Miller
    - Wrote the Section 1: Introduction. Cleaned up Section 2 from previous paper.
    - Wrote Section 3.1.
    - Created 4.1 table entries for login, logout, products, create user, and update user tests.
    - Created Section 4.2, the Usability Table.
- Jacques Troussard
    - Section 3.4
    - Section 3.5
    - Section 4.1 - Items related to invoicing
    - Editing: renumbering tables
- Taylor Dohmen
    - Section 2 from previous paper.
    - Section 3.2 & 3.3.
    - 4.1 table entries for invoice search, file input, customer pages.
    - Appendix B.1.