Jacob Stotser

Project 2 & Protocol Description

Common Case #1: As a user, I want to add a new product into the system. (We can do this using "Save Product" Button)

Actor	System
Choose command "Manage Products"	2. Displays "Update Product Info" screen
Main Screen:	"Update Product Info" Screen:
Store Management System Manage Products Manage Customers Manage Purchases	Update Product Information Load Product Product ID: Name: Price: Quantity:
3. Input data then click "Save Product" Button	4. Display "Product is Saved Successfully"
"Update Product Info" Screen with Data: Update Product Information	"Product is Saved Successfully" screen:
Load Product Save Product	Product is Saved Successfully!
5. Click on "OK" button	6. Display Main Screen
Product is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Common Case #2: As a user, I want to update product information (we can do this by loading and then saving a product)

Actor	System
1. Choose command "Manage Products" Main Screen: Store Management System Manage Products Manage Purchases	2. Displays "Update Product Info" screen, input Product ID and press "Load Product" "Update Product Info" Screen: Update Product Information Load Product Save Product Product ID: 101 Name: Price: Quantity:
3. Change the desired data then click "Save Product" Button "Update Product Info" Screen with NEW Data:	4. Display "Product is Saved Successfully" "Product is Saved Successfully" screen:
Update Product Information Load Product Product ID: 101 Name: Iphone 11 Price: 999.99 Quantity: 10	Product is Saved Successfully!
5. Click on "OK" button	6. Display Main Screen
Product is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Common Case #3: As a user, I want to add a new customer into the system. (We can do this using "Save Customer" Button)

Actor	System
Choose command "Manage Customers"	2. Displays "Update Customer Info" screen
Main Screen:	"Update Customer Info" Screen:
Store Management System Manage Products Manage Customers Manage Purchases	Update Customer Information Load Customer Customer ID: Name: Phone: Address:
3. Input data then click "Save Customer"	4. Display "Customer is Saved Successfully"
Button "Update Customer Info" Screen with Data:	"Customer is Saved Successfully" screen:
Update Customer Information Load Customer Customer ID: 201 Name: John Doe Phone: 555-666-7777 Address: 103 Miami Dr.	Customer is Saved Successfully!
5. Click on "OK" button	6. Display Main Screen
Customer is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Common Case #4: As a user, I want to update customer information (we can do this by loading and then saving a customer)

Actor	System
Choose command "Manage Customers" Main Screen:	Displays "Update Customer Info" screen, input Customer ID and press "Load Customer" "Update Customer Info" Screen:
Store Management System Manage Products Manage Customers Manage Purchases	Update Customer Information Load Customer Customer ID: 201 Name: Phone: Address:
 Change the desired data then click "Save Customer" Button "Update Customer Info" Screen with NEW Data: 	4. Display "Customer is Saved Successfully""Customer is Saved Successfully" screen:
Update Customer Information Load Customer Customer ID: 201 Name: John Doe Phone: 555-666-7777 Address: 103 Miami Dr.	Customer is Saved Successfully!
5. Click on "OK" button	6. Display Main Screen
Customer is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Common Case #5: As a user, I want to add a purchase from a customer into the system. (We can do this using "Save Purchase" Button)

Actor	System
Choose command "Manage Purchases"	2. Displays "Update Purchase Info" screen
Main Screen:	"Update Purchase Info" Screen:
Store Management System Manage Products Manage Customers Manage Purchases	Update Purchase Information Purchase ID:
3. Input data then click "Save Purchase" Button	4. Display "Purchase is Saved Successfully"
"Update Purchase Info" Screen with Data:	"Purchase is Saved Successfully" screen:
Update Purchase Information Purchase ID: 301 Date of purchase: Wed Oct 16 23:05:41 CDT 2019 Customer ID: 112 Customer Name: John Doe Product ID: 203 Product Name: Apple Iphone 11 Quantity: 1 Product Price: 999.99 Cost: \$999.99 Tax: \$100.00 Total Cost: \$1099.99 Load Purchase Save Purchase	Purchase is Saved Successfully!
5. Click on "OK" button	6. Display Main Screen
Purchase is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Common Case #6: As a user, I want to update purchase information (we can do this by loading and then saving a customer)

Actor	System
Choose command "Manage Purchases" Main Screen:	Displays "Update Purchase Info" screen, input Purchase ID and press "Load Purchase" "Update Purchase Info" Screen: "Update Purchase Info" Screen:
Main Screen:	"Update Purchase Info" Screen:
Store Management System Manage Products Manage Customers Manage Purchases	Update Purchase Information Purchase ID: Date of purchase: Customer ID: Customer Name: Product ID: Product Name: Quantity: Product Price: Cost: \$ Tax: \$ Total Cost: \$
	Load Purchase Save Purchase
 Change the desired data then click "Save Purchase" Button 	3. Display "Purchase is Saved Successfully"
"Update Purchase Info" Screen with NEW Data:	"Purchase is Saved Successfully" screen:
Update Purchase Information Purchase ID: 301 Date of purchase: Wed Oct 16 23:05:41 CDT 2019 Customer ID: 112 Customer Name: John Doe Product ID: 203 Product Name: Apple Iphone 11 Quantity: 1 Product Price: 999.99 Cost: \$999.99 Tax: \$100.00 Total Cost: \$1099.99 Load Purchase Save Purchase	Purchase is Saved Successfully!
4. Click on "OK" button	5. Display Main Screen
Purchase is Saved Successfully!	Store Management System Manage Products Manage Customers Manage Purchases

Client/Server Protocol Description

This communication system is made up of two parts: a client and a server. The two parts communicate over a computer network. The server runs a program that the client can then connect to, requiring the client to first initiate communication with the server.

Typically, the client sends a request which the server then responds to. In order to maintain this communication, the client and server must follow certain rules, or "communications protocols." For our system, there is a server running that manages the database document, and the client communicates through a port (port 1000 in our case). The client sends requests, including the following:

- GET a product
- PUT a product
- GET a customer
- PUT a customer
- GET a purchase
- PUT a purchase

When the client sends any of these requests, the server processes the information and updates the SQL database file. For example, consider the user pressing the "Load Product" button on the graphical user interface. The client would gather the information from the GUI (ProductID) and send a GET command to the server, which the server would then process and send the product information back to the client. This process is repeated, with slight variation, for the other commands in this project.