CIS 304
Project 3
Dr. Curl

SmartStore.com

Creating, Customizing, and Configuring an Applet
150 points
Includes Extra Credit

Purpose: This problem emphasizes the design and coding of an object-oriented application using a graphical user interface and an applet. Java graphical swing classes, event handling, and an applet class are used to design and implement a user friendly interface.

Problem

This project extends your previous work by running your application inside of a web browser. To provide further flexibility, the applet's html file will include customizing tags. Note that all input and output by the user will continue to come only from the graphical interface developed in Project 2.

Applet

Convert your existing Java application to an applet. Create an html file to hold your work and test it by running locally on your machine. When complete, create a folder name "CIS304" in your Cal Poly Pomona web account and test your work running over the Internet. Be sure to allow me access to this folder by granting appropriate permissions (i.e., public or scurl). *Important! Be sure to clearly identify the URL of your applet for testing in the comments section when submitting your project.*

Customization

Modify your applet so that the user can include customized settings via the web page using "param" tags. Include the following tags in your html page: your name, CIS 304, plus three other customizing features of your own choosing (e.g., font family, font size, background image, etc.). All parameters should be functional within the application. *Your program will only run when the first two parameters are included correctly; otherwise, display an appropriate error message.*

File Processing

All information pertaining to items must be read from an XML file. Be sure to remove these values from the program and save them as a file named items.xml. The file structure is left for the student to decide.

Test Data and Calculations

Using the sample data provided, build the following abbreviated table of test results. Note these are only for good inputs. Your program should continue to check for bad inputs. In the last column, indicate Yes or No, as to whether or not your program's output agrees with expectations.

Test Table and Results

Test no.	Test	Description	Correct (Y/N)
1.	List	Listing of items for sale with counts	
2.	Details	Show details for item PL	
3.	Add	Add items PL, HWGA to cart	
4.	Add	Add another item HWGA to cart	
5.	Cart	List of items in cart	
6.	Return	Return item HWGA from cart to inventory	
7.	List	List of items for sale with counts	
8.	Help	Show menu of commands	

Submit:

1. Program source code with appropriate formatting and comments.
2. Completed table of test results titled *Project 3 Test Results.*
3. Input/output dialog of your graphical user interface applet *showing the web page with the www.csupomona.edu address* (using screen prints) for a complete set of test cases. Be sure to label what is being shown.
4. Configuration text file.
5. HTML file.

All of the above must be contained in a jar file named XYZProject3.zip, where XYZ will be your initials, and submitted through Blackboard by the due date. Be sure to check your files to see that you have included everything before submitting your work.

Grading

	Points
– Program code	90
– Test table	15
– Dialog	15
– XML file	15
– HTML file	15
Total	150

Extra Credit (10 points)

Organize your work into two packages: all user interface classes should be placed into a package named *gui* and all business classes placed into a package names *business*. Be sure to use lower case for both package names. Deploy your Java work as a JAR file named SmartStore.jar and modify your web page to recognize both the your use of packages and the Java archive file. Add "Includes deployment extra credit" in the Blackboard comments section when your project is submitted.