Class: CSC 4350

Semester: Spring 2017

Project Title: Online Shopping System - Stop & Shop

Group Name: DMD (Team 9)

Members:

Dwight Martin

Mechal Terfie

Dalya Khatun

Date: 04-23-2017

Table of contents:

DMD- 1.0 Introduction:

Our project is an online shopping system. Online shopping is a form of electronic commerce which allows consumers to directly buy goods or services from a seller. Our shopping system will have three modules: customers (buyers), employee (sellers), and admin. Each of them will login to different environments to perform their tasks.

CUSTOMER: The system shall allow Customers to register, login, search, view, select add products, purchase products, and make payments.

EMPLOYEE: The system shall allow Employees to register, login, post products, update products, and distribute products.

ADMIN: The system shall allow admins to login, generate reports, and display products.

2.0: CAPTURE "SHALL" STATEMENTS

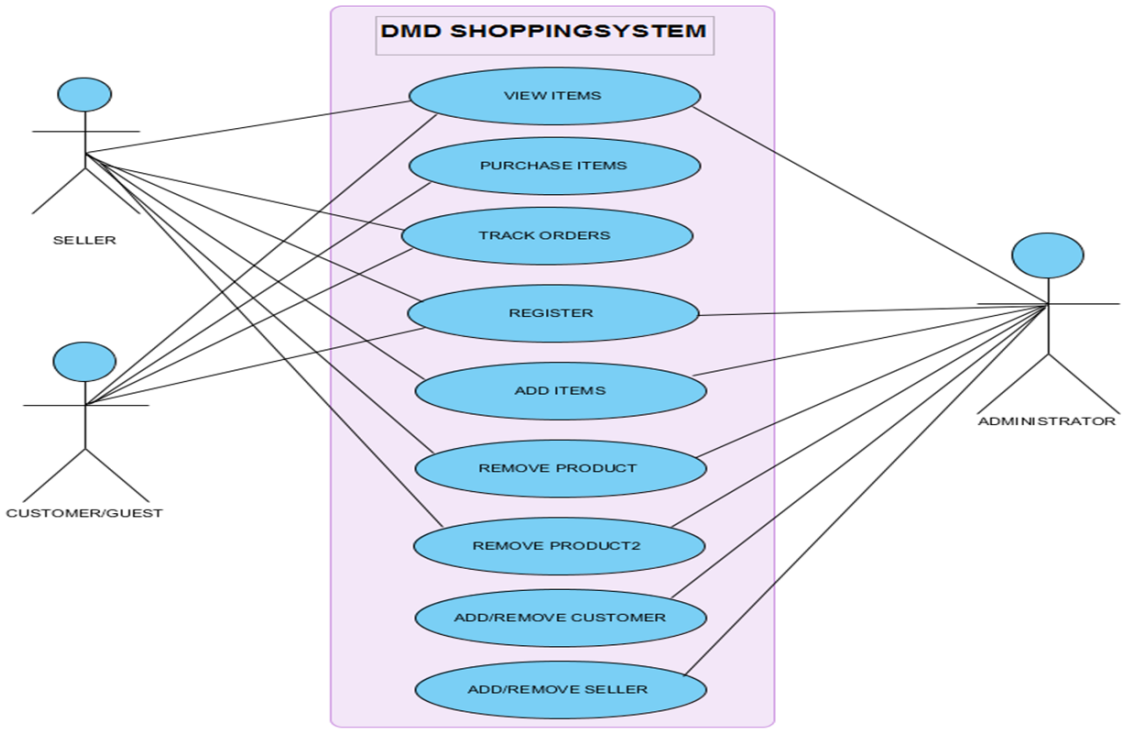
2.0 Customer	3.0 Employee	4.0 Admin
2.1 The system shall allow Customers to register and login.	**3.1 The system shall allow Employees to register and login.**	**4.1 The system shall allow admins to login.**
2.2 The system shall allow customers to search and view products.	**3.2 The system shall allow employees to post, remove and update product.**	**4.2 The system shall allow to have total permission to do anything with the website.**
2.3 The system shall allow customers to select and add product to shopping cart.		**4.3 The system shall allow to generate report.**
2.4 The system shall allow customers to make purchases and receive products.	**3.3 The system shall allow employees to distribute product.**	**4.4 The system shall allow to display report.**

5.0: RTM

Entry #	Para #	System Specification Text	Type	Build
1	2.1	The system shall allow Customers to register and login.	SW	B1
2	3.1	The system shall allow Employees to register and login.	SW	B1
3	4.1	The system shall allow admins to login.	SW	B1
4	2.1	The system shall allow customers to search and view products.	SW	B1
5	2.3	The system shall allow customers to select and add product to shopping cart.	SW	B1
6	3.2	The system shall allow employees to post, remove and update product.	SW	B1
7	4.2	The system shall employees allow to have total permission to do anything with the website	SW	B1
8	4.3	The system shall allow admin to generate report.	SW	B1

9	2.4	The system shall allow customers to make purchases and receive products.	SW	B1
10	2.5	The system shall allow customers to make payments with different payment methods.	SW	B1
11	3.3	The system shall allow employees to distribute product.	SW	B1
12	4.4	The system shall allow to display report.	SW	B1

6.0: Use Case


DMD SHOPPINGSYSTEM
VIEW ITEMS
PURCHASE ITEMS
TRACK ORDERS
REGISTER
ADD ITEMS
REMOVE PRODUCT
REMOVE PRODUCT2
ADD/REMOVE CUSTOMER
ADD/REMOVE SELLER
SELLER
CUSTOMER/GUEST
ADMINISTRATOR

7.0: Rational for use cases

The following class shall perform the functions as described.

Register:

If the customer or employee is a new user, he/she can request to register with the system. The system displays a registration page and asks the customer or employee to choose a login name (email address) and password. The customer is also required to enter their name and address.

Login:

The customer or employee can login to the shopping system with user name and password. The system will verify that the login name matches the login password. If they do not match, error message will be indicated.

Search Product:

The customer enters product search parameters and requests a product search. The system will search through the products category in its database and return the matches to the customer. If there are no matches, the system will display a fail message.

Select and add product:

When customers find the products they want, the system will allow them to add products to the shopping cart. Store and keep track of the information of the products that have been added into shopping cart. Allow the customer to view the contents of the shopping cart.

Confirm order:

When the customer requests to checkout, the system will prompt credit card information page. The customer will be given a choice on whether he/she wants the item shipped to stored address or to an alternative address. The input payment information will be save into the order form.

Confirm payment:

When the customer checks out, the credit verification company validates the customer's credit card when given the customer's name, credit card number, and expiration date, and then returns the validation result to checkout department (Employee/Seller). If the response shows that the credit card is invalid, the customer will be asked to re-input his payment information.

Order details:

Allow customer to view the product they purchased and will be able to see the tracking information.

Maintain product:

The employees request to update products information. This includes the products price, description, brand, title, or number, adding new products, removing products. The system will save the updated product information in the database.

Generate report:

This class will generate sales report, product report.

8.0 UC01_DMD_Online_Shopping_System_**CID**

1. User logs into Site Interface.

2. Site Interface displays login verification to User.

3. Site Interfaces displays customer menu to User.

4. User generates search request to Site Interface.

5. Site Interfaces searches inventory in Database.

6. Database retrieves inventory and sends to Interface.

7. Interfaces displays inventory to User.

8. User selects item(s) from inventory displayed by Site Interface.

9. Interfaces sends item(s) to Shopping Cart.

10. Shopping Cart sends updated cart to Site Interface.

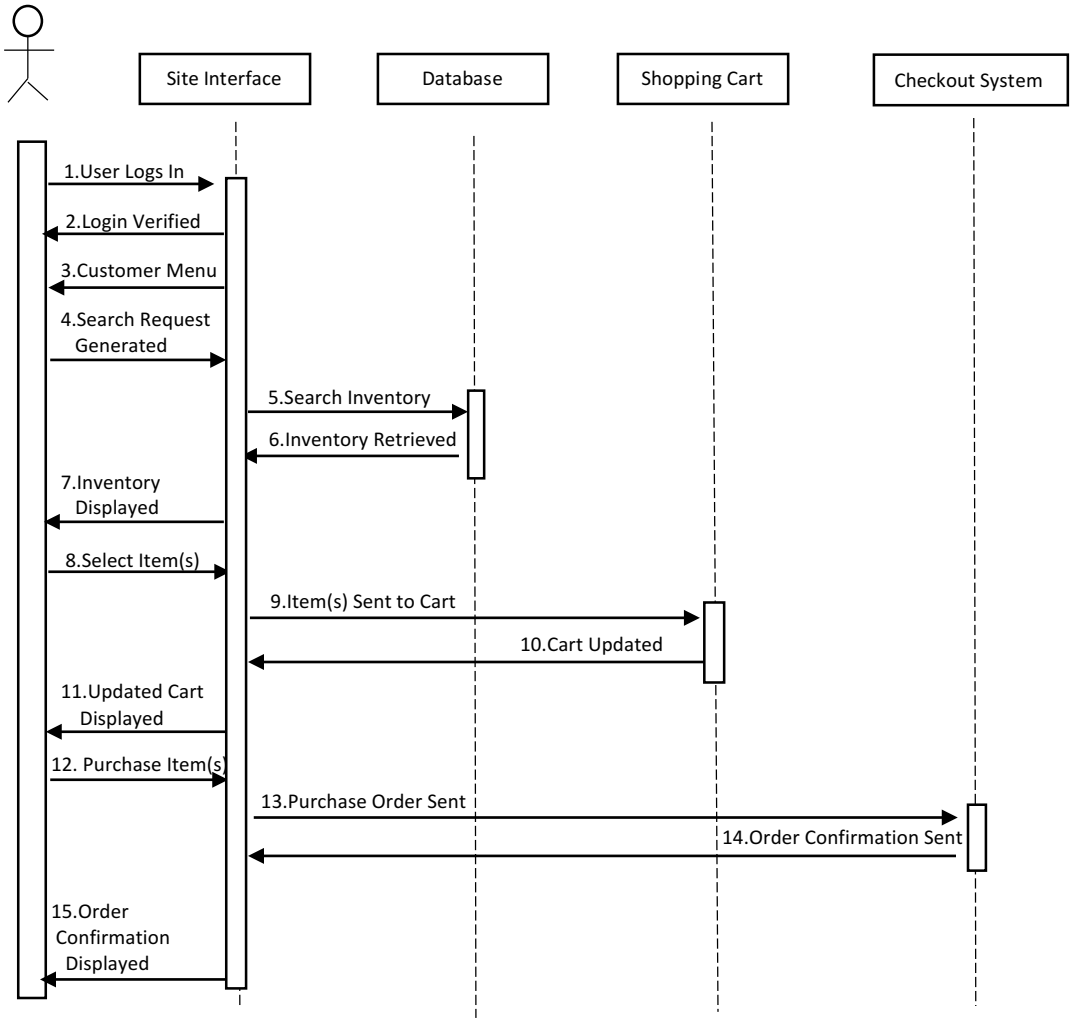
11. Site Interface displays updated cart to User.

12. User select to purchase item(s) on Site Interface.

13. Site Interface sends purchase order to checkout system.

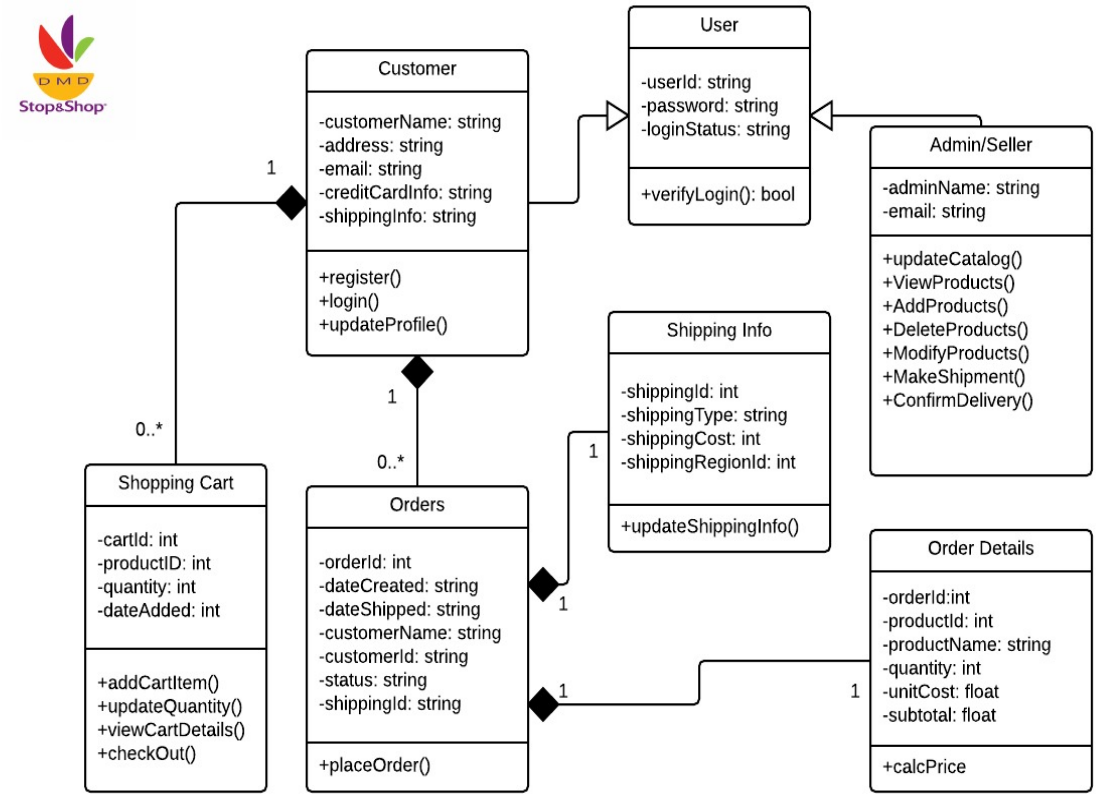
14. Checkout System sends order confirmation to Site Interface.

15. Site Interface displays confirmation order to User.



9.0 Rationale for Objects (Object Design)

DIAGRAM SHOWING THE CLASS INTERFACE OF THE DMD SHOPPING SYSTEM



Data Objects

- User: user ID, Password, login Status
- Customer: Customer Name, Address, Email, Credit Card Info, Shipping Info
- Admin/seller: Admin Name, Email
- Shipping: Shipping ID, Shipping Type, Shipping Cost, Shipping Region ID
- Shopping Cart: Cart ID, Product ID, Quantity, Date Added,

- Orders: Order ID, Date Created, Date Shipped, Customer Name, Customer ID, Status, Shipping ID
- Shipping Info: Shipping ID, Shipping Type, Shipping Cost, Shipping Region ID
- Order Details: Order ID, Product ID, Product Name, Quantity, Unit Cost, Subtotal

User

This store information about both the sellers and buyers which includes attributes such as user Id, password and login status.

Customers

This stores information about the user including attributes such as Customer Name, Address, Email, Credit Card Info, Shipping Info.

Admin

This manages the data and information of both seller and buyers both sellers, both of which has certain restriction in terms of information accessibility which includes; Names and emails addresses.

Shipping

This manages the retrieval of information about customer's order and shipping information such as; Shipping ID, Shipping Type, Shipping Cost, Shipping Region ID.

Shopping Cart

This provides information about the customer's item that is intended to be purchase. These information includes, Cart ID, Product ID, Quantity, Date Added.

Orders

This provides information about the customer's past transactions, or current/pending transactions. These attributes include; Order ID, Date Created, Date Shipped, Customer Name, Customer ID, Status, Shipping ID.

Order Details

This also holds/provides information similar to the attributes mention prior from Orders, also provides extra information such as Product ID, Product Name, Quantity, Unit Cost, Subtotal.

10.0 Test Cases:

Test Case Number	Test Case Methods Name	Test Case Description	Functional requirements	Expected result
TC01	CheckAuthentication();	To test the Login/Authentication for the Customer, Seller and Admin.	FR01	Pass
TC02	viewShoppingCart();	To test, users can view the items they add in the shopping cart.	FR02	Pass
TC03	AddUpdateRemoveProducts();	To test, Customer can add upload and remove products.	FR03	Pass
TC04	checkValidOrderInfo();	To test that customer are not able to submit an order form if the information in any of the fields is not valid.	FR04	Pass
TC05	CheckMissingInfo();	To test that customer are not able to submit an order form if the information in any of the fields fields is left blank.	FR05	Pass

TC01: To test the Login/Authentication interface

- Input: Username and Password
- Output: Valid Destination Page

- Valid Range: User Name →Alphanumeric, Password → Alphanumeric
- End Messages/Result

If (User == Valid User), an order form appears to complete the checkout process

i. If (User != Valid User), an error message is displayed on the Login interface.

TC02: To test, the users can view the items they add to the shopping cart.

- Description of Purpose: The system shows all the saved items in shopping cart for a particular user. The user can choose to check out the items or go back to continue shopping.
- Input: The user adds an item to the shopping cart from any of the available categories.
- Output: The shopping-cart page pops up, showing the item that is added by the user.
- End messages/Result
  i. If (Selection == Item and document == exists), the user is able to add that item to the cart, and the item shows up in the shopping cart, prompting user to delete the item, to continue shopping, or to check out the item.
  ii. If (Selection = Item and Selection = View Cart), an empty shopping cart pops up with buttons to check out or to continue shopping.

TC03: To test, Seller can add upload and remove products.

- Input
  i. User=Admin
  ii. Selection=Items
- Output: New or modified items in the shopping cart.
- End messages/Result

  i. If (User type = ="Customer" && Selection = Items) , then display the modified items in the shopping cart.

TC04:

- Input

  i. Shipping address
  ii. Payment method
  iii. Name
- Output

Order confirmed

If (the_any_fields==valid) {

Ask the customer to correct the information.

}

TC04:

- Input
  i. Shipping address
  ii. Payment method
  iii. Name
- Output

  Order confirmed

  If (the_any_fields==null) {

  Ask the customer to enter the missing information.

  }

Functional requirements description:

FR01: The DMD shopping system shall have three types of authentication: Customer authentication, seller authentication and Admin authentication.

FR02: The users shall be able to view the items they added to the shopping cart.

FR03: The customer shall be able to add, update and remove items from the shopping cart.

FR04: The customer shall not be able to place an order without providing valid information for all rows in the order form.

FR05: The customer shall not be able to place an order if any of the columns in the order form are left empty.

11.0 Rationale for Test cases

TC01. CheckAuthentication():

The check Authentication test for the customer's, seller's and admin credentials to make sure the username that is enter matches password that is stored in the database.

TC02. viewShoppingCart(): This test case allows the users including the administrator to access items that are added to the cart, this also facilitates product updates by adding more items, increasing and decreasing the quantity that is intended to order.

TC03. AddUpdateRemoveProducts(): This gives the ability to the user to view and update his or her cart. Some of the functionalities includes, add a product to the cart, update quantity of that product or even removing that product.

TC04. checkValidOrderInfo();

This test case involves a form where all users of the system have to enter their information that will be stored in the database and later be used in order to completely process a purchase. For example, the have to enter they production information and also the address from which the product will be shipped from. In addition, the customer/buyer would also have to enter their information in, so that the order can be process successfully. This information would be tested for validity and will include email address, shipping address name etc.

After filling in the information, the user places the order by clicking on the place button, bringing him to a final message page indicating that the order was successfully placed.

TC05. CheckMissingInfo ();

To test that customer are not able to submit an order form if the information in any of the fields is left blank.

11.0 Rational for DMD

Our project is an online shopping system. Online shopping is a form of electronic commerce which allows consumers to directly buy goods or services from a seller. The flow begins when the customer first runs the application home screen online shopping system application that appears in the web browser. User can select to be a customer, seller, or admin. Customer and seller can register and login. The Customer can browse through the available list of categories and can choose either to select a category or to directly view the cart.

In the category, a user can select view more information for details about a particular item before deciding to add it to the shopping cart by clicking on the cart icon next to the item. The user can then decide to either continue shopping by clicking the continue shopping button or can check out by clicking on the checkout option. If there are no items in the cart, then the user does not have an option to click checkout. The user can check out after doing the user authentication by logging in with the username and password. Once the user successfully logins/registers, the order form, where the user can put the correct information to place the order appears.

If the user includes incorrect or incomplete information, then placing the order is not allowed. After the user successfully inputs the correct information, placing an order is successful and the user can see the "success" message. The additional flow step for the administrators is that they can view the user's information, the user's checkout, and the product details by using the database after the user successfully places an order. They can view the the orders that customer placed and confirm in shipment.

12.0 COCOMO II – Constructive Cost Model

COCOMO II - Constructive Cost Model

Software Size Sizing Method: Source Lines of Code

	SLOC	% Design Modified	% Code Modified	% Integration Required	Assessment and Assimilation (0% - 8%)	Software Understanding (0% - 50%)	Unfamiliarity (0-1)
New	5000						
Reused	2500	0	0	30	3		
Modified	2500	30	30	50	5	25	0.2

Software Scale Drivers

Precedentedness	Low	Architecture / Risk Resolution	Nominal	Process Maturity	Low
Development Flexibility	Nominal	Team Cohesion	High		

Software Cost Drivers

Product		Personnel		Platform	
Required Software Reliability	Nominal	Analyst Capability	Nominal	Time Constraint	Nominal
Data Base Size	Nominal	Programmer Capability	Nominal	Storage Constraint	Nominal
Product Complexity	Low	Personnel Continuity	Nominal	Platform Volatility	Low
Developed for Reusability	Nominal	Application Experience	Nominal	**Project**	
Documentation Match to Lifecycle Needs	Low	Platform Experience	Low	Use of Software Tools	Nominal
		Language and Toolset Experience	Nominal	Multisite Development	Nominal
				Required Development Schedule	High

Maintenance Off

Software Labor Rates

Cost per Person-Month (Dollars) 0

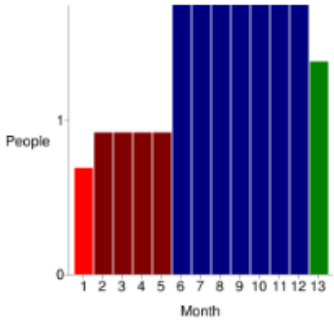
Results

Software Development (Elaboration and Construction)

Effort = 17.6 Person-months
Schedule = 12.3 Months
Cost = $0

Total Equivalent Size = 6415 SLOC

Staffing Profile



Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.1	1.5	0.7	$0
Elaboration	4.2	4.6	0.9	$0
Construction	13.4	7.7	1.7	$0
Transition	2.1	1.5	1.4	$0

Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.1	0.5	1.3	0.3
Environment/CM	0.1	0.3	0.7	0.1
Requirements	0.4	0.8	1.1	0.1
Design	0.2	1.5	2.1	0.1
Implementation	0.1	0.5	4.5	0.4
Assessment	0.1	0.4	3.2	0.5
Deployment	0.0	0.1	0.4	0.6

Your output file is http://csse.usc.edu/tools/data/COCOMO_March_27_2017_02_31_09_101452.txt

Created by Ray Madachy at the Naval Postgraduate School. For more information contact him at rjmadach@nps.edu

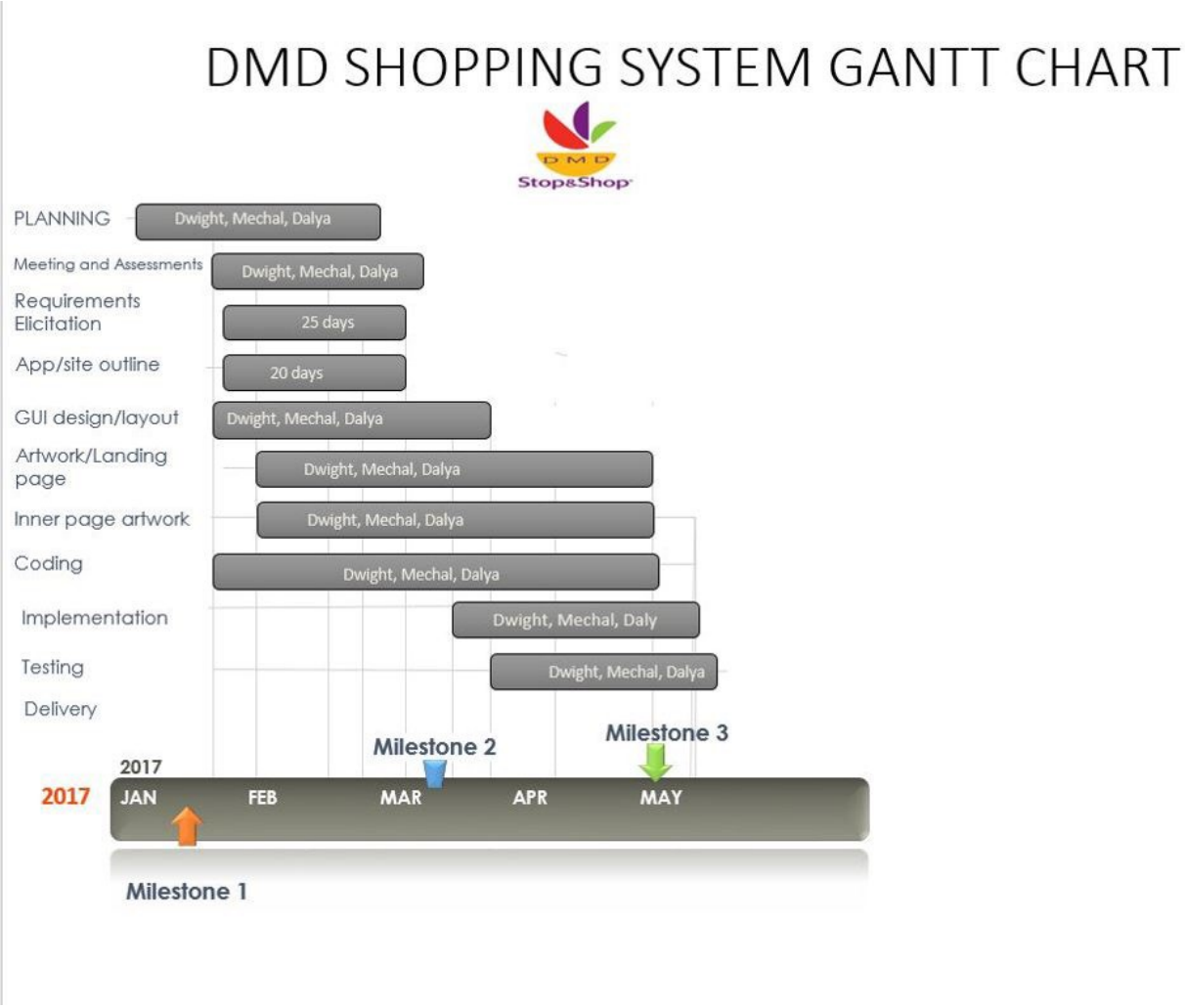
13.0 Project Legacy

Our project legacy began when we started this semester at January. We learned important lessons over the course. It was difficult when we first started this project. We had plenty of resources but it was confusing to narrow down to the one that we actually could use for this project. We wish there was more instructions on choosing the tools. We did our project using java swing but we started with javaFX. We lost first month exploring and experimenting different possibilities. We could've make our project better if we had more ideas about deferent perspectives of java when we first started the project. We spent more time starting the project then actually functioning the project. End of this project was worthy because we learned a lot and next time we develop a software we will know where to start.

14.0 WSD FOR DMD (TEAM 9)

Task #	Name	Task
1	Mechal Terfie	Team coordinator
2	Mechal Terfie	Documents Handler, documentation
3	Mechal Terfie	Java coder, admin page
4	Dwight Martin	GUI tester
5	Dwight Martin	Finalize code, documentation
6	Dwight Martin	Java coder, customer page
7	Dalya Khatun	User Guide, documentation
8	Dalya Khatun	Java coder, seller page

15.0 Gantt Chart



16.0: Dictionary/Glossary

Abstract Data Types (ADT) A type whose internal form is hidden behind a set of access functions. Objects of the type are created and inspected only by calls to the access functions. This allows the implementation of the type to be changed without requiring any changes outside the module in which it is defined.

Applications Programmer Interface (API) The interface (calling conventions) by which an application program accesses operating system and other services. An API is defined at source code level and provides a level of abstraction between the application and the kernel (or other privileged utilities) to ensure the portability of the code

Architectural Design Large systems are divided into smaller subsystems and modules which import from each other. The subsystems and modules and their use relationship is called the architectural design.

Class A class defines an software object's interface and implementation. It specifies the object's internal representation and defines the operations that the object can be instructed to perform.

Database One or more large structured sets of persistent data, usually associated with software to update and query the data. A simple database might be a single file containing many records, each of which contains the same set of fields where each field is a certain fixed width.

Deliverable this is specific work product, such as requirements or design documentation, produced during a task or activity to validate successful completion of the task or activity.

Design Document A less formal term software Design Specification (SDS).

Encapsulation The result of hiding a representation and implementation in an object. The representation is not visible and cannot be accessed directly from outside the object. Operations are the only way to access and modify an object's state. [james93]

Lifecycle this is a set of Software development activities, or stages that function together to guide the development and maintenance of software products.

OMT Model A model built with the Object Modelling Technique. An OMT model may have three views (object, dynamic and functional). It consists of a set of diagrams and associated information necessary for characterizing the domain (e.g., data dictionary, modelling rationale, costs, etc.)

Reverse Engineering The process of analyzing an existing system to identify its components and their interrelationships and create representations of the system in

another form or at a higher level of abstraction. Reverse engineering is usually undertaken in order to redesign the system for better maintainability or to produce a copy of a system without access to the design from which it was originally produced.

Software Design Document (SDD) this is a document that defines the storage mechanisms, interfaces, internal processes and outputs of a software application.

Subsystem An independent group of classes that collaborate to fulfill a set of responsibilities.

Use case is a list of actions or event steps, typically defining the interactions between a role and a system, to achieve a goal.

User Interface, Graphical (GUI) The use of pictures rather than just words to represent the input and output of a program. A program with a GUI runs under some windowing system (e.g. The X Window System, Microsoft Windows, Acorn RISC OS, NEXTSTEP).

Software Engineering A systematic approach to the analysis, design, implementation and maintenance of software. It often involves the use of CASE tools. There are various models of the software life-cycle, and many methodologies for the different phases.

System is a collection of hardware, software, firmware, and documentation components organized to accomplish a specific function or set of related functions.

System Testing is the analysis of a problem that the organization will try to solve with an information system.

Task is the smallest accountability unit of work. A task is the lowest level of work division typically included in the Project Plan and Work Breakdown Structure.

Test Case is a defined set of database records, test inputs, execution conditions and anticipated results designed to exercise specific application components and verify compliance with design criteria and requirements. Contains detailed instructions for the setup, execution, and evaluation of the result for the test case.

Use Case is a description of a business process under automation, focused on how Actors (users and interfacing systems) interact with the process. Includes descriptions of the information the Actors send to the system, the data the Actors receive from the process, and the operations they perform using the system.

17.0: Resumes

DWIGHT L MARTIN
4050 SOUTHWOOD CIR, APT 1113 ATLANTA, GA 30331 Tel:678-724-7889 Email: dwight2k9@yahoo.com

OBJECTIVE

Energetic and career-minded individual with an academic record that reflects responsibility and leadership. One who is in active involvement in the school community while offering disciplined work habits, and high level of initiative. In addition, demonstrate ability to balance competing demands.

Knowledge of:
- MAC OSX
- Microsoft Windows XP/Vista
- Win7/Win8/Win10, UNIX, Linux
- JAVA, JavaScript, HTML, CSS, PHP

Extensive experience in:
- PC/MAC hardware /application
- MS Office 2007/2010/2013/2016
- Google App
- Adobe InDesign, Photoshop MySQL

EDUCATION

2014 – expected December 2017 — **Bachelor of Science in Computer Science**, Georgia State University, Atlanta GA

2008 -2010 — **Associate Degree in Natural Sciences**, University of Technology (UTECH), Jamaica

2004 – 2008 — **Diploma in Engineering Technology**, Dinthill Technical High School, Jamaica

SEMESTER PROJECTS at GEORGIA STATE UNIVERSITY 2014-CURRENT

- ECommerce website, JavaScript, CSS, HTML, PHP 4.1, MySQL
- Dictionary, JAVA, Data structures.
- **Data Structures:** File Indexing using Binary Trees in Java.
- Contact Phonebook, C/C++, System level programming
- Online Shopping Cart, Java Swing, SQLite, JavaScript (Software Engineering)
- Graphical User Interface, Java, OOP (Inheritance, Polymorphism etc.)
- Design Analysis & algorithm (searching, sorting, etc.)
- **FTP Client,** Linux, Socket Programming, C.

EXPERIENCE

2013 - Current — **Gourmet Assistant Manager & Team Lead**, Atlanta Oriental Foods Wholesale Co.
- Oversee activities, efforts, and training of 12 team associates and effectively achieving company's goals
- Proficient with MS Office (Word, Excel, Access, PowerPoint), Store Management Suite Software, various Internet search engines, and e-mail programs
- Ordering, Buying, Labelling, and Adding in new items in SMS database
- Arranging stock to reassure that items do not exceed the expiration date

2012 - 2013 — **Cashier & Inventory Specialist**, Bargain Book Warehouse, Wynnewood, PA.
- Having a good knowledge of the products that are available.
- Assisted with processing customer orders, cleaning, and stocking
- ***2012 Employee of the Year Award in recognition of 55% sales increase.***

LEADERSHIP/ACHIEVEMENTS

- Dean Honor List (SGPA > 3.5) 1st to 4th semester
- Public Relation Officer (PRO) of the Interact Club at University of Technology Jamaica
- ***2012 Employee of the Year Award in recognition of 55% sales increase***
- Member of the Winning squad of Inter-House Cricket Tournament at University of Technology Jamaica (UTE

CO-CURRICULAR ACTIVITIES

- Member of Science Club at University of Technology Jamaica
- Captain of the School of Arts and Science Faculty Soccer Team
- Executive Member of the Association "Each One Teach One" Tutoring at University of Technology Jamaica

Name/introduction

My name is Mechal Terfie. Currently working towards a bachelor's degree in Computer Science with a concentrating in Database and Knowledge-based systems. I am expected to graduate the fall of 2017.

Schools

Georgia Perimeter College

Georgia State University

Classes taken

- Java 1&2
- System level programming, C basic
- Data structures, Java
- Computer organization and programming, SPARC assembly
- Introduction to Matlab programming

Currently

- Software engineering
- Operating systems
- Programming language concepts
- Database systems.

Skills

- 3 years' java experience
- 1-year C basic experience
- 6 months' assembly experience
- 5 months' Matlab experience

Achievements

Received an opportunity to work in the field of data entry at Fox Environmental LLC. Duties entailed record keeping in excel data form, coding information, and insuring information organization and confidentiality.

DALYA KHATUN

Dalya727@yahoo.com

OBJECTIVE

To learn how to make software in java.

CLASSWORK

- Intro to computer science
- Principle of computer science
- Data structure
- Computer organization
- System level programming
- Computer architecture
- Web development
- Calculus I
- Calculus II
- Discrete Math
- Math model of computer science
- In Progress: Database System, Data Mining, Programming Language Concepts, Software Engineering, LT.

SKILLS

- Java,
- html,
- CSS,
- JavaScript,
- Unix shell scripting.

AWARDS AND ACKNOWLEDGEMENTS

- President's Award for Educational Excellence.

18. Prototype

Initial requirements: Develop a software that allows users to buy, sell or maintain a shopping system.

Initial prototype:

- Three module for each user - seller, customer and admin.
- Seller should be able to add, remove, update products. View orders and confirmed shipment with email.
- Customer should be able to search product, view products, add products to shopping cart, update shopping cart and checkout.
- Admin should generate reports, delete seller and customer if necessary.

Things we could not achieve:

- Search function and payment function for customer module.
- Confirmation email function for seller portal.
- Generating reports and analytics of sales, corresponding dates from the calendar with reminders, build interactive message system between users.

19.0 User Guide

Seller Portal

How to get access to the seller portal?

- To be a part of the seller portal register as a seller and login.

How to add product?

- To add products, fill up all the information about product in seller homepage. Click **add product** and your product will be added to DMD shopping system.

How to view the products that I added?

- To view products, click on "My Products" on seller home page.

How to update product?

- To update product, click on my product first and note the product ID that you want to update. Enter product Id on top of "Update product" button. Fill up all the product information on the left then click "Update Product". Click "My Product", Your product should now be updated.

How to delete product?

- To delete product, enter product ID and click on "Delete"

How to view Orders?

- To view orders, click on shipment on seller homepage. Click on "View Orders", You will see the list of new orders on the left.

How to confirm shipment?

- Enter product ID then click "shipped" and "Confirmed".

Customer Portal:

- Enter customer portal as a guest or login. New user can register by clicking on "Register".
- Click on "View Product" to view available products.
- To add product to shopping cart, select product and click on "add to shopping cart".
- To view your cart, click on "view shopping cart" on top right corner.
- To update product, click on "update" and fill up the information and enter.
- To checkout click on "Checkout".

Seller Portal:

How to view products?

- Click on **view products** on admin portal home.

How to view Seller?

- Click on **view Seller** on admin portal home.

How to view Customer?

- Click on **view customer** on admin portal home.

How to view net sales?

- Click on "Total sale"

How to remove seller?

- Click on view seller, note the ID of the seller. Enter seller ID on top of remove seller button. Click on "Remove Seller".

How to remove customer?

- Click on view customer, note the ID of the customer. Enter customer ID on top of remove customer button. Click on "Remove Customer".