JAVA SHOPPING PROJECT

LINK TO GITHUB: <https://github.com/bayquiz/4365_Proj.git>

The goal of the project was to create a shopping console application. We wanted to give the user an option to view products, add them to their cart, and buy them. Within this process, we also added payment verification and a store bank account so users can add money to their store account.

Potential users of this product are Texas Tech students. We are selling books, so students are our main target.

VIEW PRODUCTS- Here, a user can buy an item from the list of items we have in inventory. The user can also check the amount left in his wallet and all the items remaining in the inventory. The way it works, when a user checks what we have in the inventory, he will see each item with the corresponding price and quantity. He can choose what to do from a menu item. If the account balance (wallet) is less than the total amount he wants to spend, the application will return an error message when the user views their cart.

FILTER PRODUCTS- The user can filter down the products and view by price. Once all products are displayed, the user can choose to filter products less than $50, and less than $25. From here, the user can continue to shop or add an item to their cart.

ADD TO CART- After viewing the list of products, the user can add an item to their cart. They do this by typing the product id of their desired item. This utilizes UI.java and Cart.java. More specifically, it uses the addProductToCart, and addProductToCartByPID methods. Both take the user input and add the assigned PID product, to the cart.

VIEW CART- When the user views their cart, they can choose to remove and item (by PID), add an item (by PID), checkout, or return to the main menu. All the user interface occurs in UI.java, but each of the menu actions is pulled from other files. View cart also displays the cart total, and the user’s store account total. The cart gives a warning message if the users store account is less than the cart total.

REMOVE FROM CART- Once the user views their cart, they may want to remove an item. This uses removeProductFromCart and calls a function in the cart class that removes an item by its unique id number.

CHECKOUT- This really combines the credit card verification and the store bank account. Here, the user can check out with all their items in their cart. If the user has a store account, and the balance is greater than the cart total, the process is already complete, and money will be deducted from their store account. If the user does not have a store account, they simply checkout with their credit card. If the user does have a store account, but the funds are not sufficient, they will checkout with their credit card for the remainder of the funds.

CREDIT CARD VERIFICATION- The Payment Verification case is used for the verification of credit cards using the Luhn Algorithm. The Luhn algorithm formula is used to validate a variety of Credit card numbers and verifies a number to generate the full account number and must pass an algorithm test to be verified. The user will be able to input their credit card number and it must follow the algorithm or the card will be declined, and the user will be put into a loop until they enter a valid card number. Once the card number is accepted the user will have to input their CVV and it must be a three-digit number, or it will throw an error and the user will have to repeat the input until it’s a three-digit number is entered. After the credit card number and CVV are entered the user will input the month and year in a format used by most American credit cards, (MM/YYYY) and if done correctly the credit card will be accepted letting the user load the card into their bank account showing the balance of the card. Once seeing the balance, the user will be able to determine what products they want to purchase in the online store.

BANK ACCOUNT / STORE ACCOUNT- The bank account class was designed to allow users to create a bank account with their name, the account id, and the initial balance in their account. Many methods were created in this class but only three are the most important. The display all, the deposit, and the withdraw. In the display all method, the system allows the user who created his account to view all the information about his account. In the deposit method, the user can add money to his original balance. Also, in this method a payment verification created earlier has been added. It asks the user for the credit card information before depositing money in the bank account. Finally, in the withdrawal method the user can withdraw the money in his account. The user can withdraw any amount of money he wants if his account can handle it otherwise, the system will give an error message.

Who did what:

Our use cases really required our work to be combined. So, this is a general idea of who accomplished what.

Saddam- Store bank account

Edmond- view products and checkout

Robert- Payment Verification (Credit Card)

Mary- Product set up

Baylee- Store menu and cart. Implemented groups work together.