CIS30A Project Documentation: Program Description

1. Noel Perez
2. Project Information and details:
    - What problems are you solving in this project?

In my project I am trying to implement an e-commerce application that simulates a customer choosing items they would like to purchase from an instrument store and adding them to their shopping cart. After the customer chooses what they would like to purchase, the shipping, tax rate and grand total are calculated for their purchase. The user can then choose a delivery date and time to have their items delivered. I am trying to create an intuitive, user-friendly, program that is as realistic as possible to an e-commerce website like amazon, or any other website that sells merchandise. Typically, this involves heavy use of GUIs and abstraction. There are also other challenges to think about like customer location for tax purposes and feasible delivery times. The logistics of delivery are important for the profitability of the business.

- What solutions are you implementing in the project?

My solutions include a shopping cart, that the customer/user can easily add items to, view it, remove items, and calculate the appropriate rates such as taxes, shipping, and grand total. The user can choose their delivery date and time with an easy-to-use GUI calendar widget. The program does exception handling so that the user has a pleasant time using the program and experiences something similar to many of the e-commerce websites that are out there today.

- Provide explanation of algorithm implementation.

The main algorithm is the menu that drives the program and executes the correct commands based on the menu choice the user chooses. Assuring that the user can only add items that are available with exception handling is crucial to a program like this. There are mostly a lot of for loops and if statements throughout the program to catch any user errors and execute the correct calculations. Assuring the correct totals are calculated is implemented with an algorithm that calculates this based on the item and its quantity.

- What are the program objectives?

The objectives of the program are for the user to pick items from a list of items for sale, along with the quantity and add them to a shopping cart. The shopping cart should have the functionality that most online ecommerce websites have, which is adding items to your shopping cart, viewing what's in your cart, a subtotal of what's in the cart and the option to remove items you don't want. My cart accomplishes all of that. Another objective is to have a proper way to "check-out" with the items that are in your cart by putting in the customer's personal information so the purchase can be delivered. After

that a receipt must be generated and written into a text file called “receipt.txt” that serves as summary of the entire transaction and proof that the order was placed. The customer must also get to choose their delivery date and time.

- Explain how your program is interacting with the user and its purpose.

The program is interacting with the user by first prompting them with a menu. The menu has 5 options in it: shop(add items to shopping cart), remove items from the cart, view what’s in the cart, checkout(finalize order and pick delivery details), and lastly exiting the program. Most of the program is dependent on the fact that there must be something in the cart for the rest of the functionality to work, so for example, the user cannot view an empty cart, nor can they remove items if there are none in it and lastly they cannot checkout if their cart is empty. Once a user has something in their cart then they can choose to view their cart and a summary of the items appears on the console. If they choose to remove an item they are prompted to choose the item and quantity they want to remove. They can only remove valid items and quantities that are in the cart, meaning if an item is not in the cart or if they choose to remove a quantity greater than what is currently in the cart then they will be prompted to pick different values. When they choose to check out, they enter their personal information, which is later echoed back to be verified and then a calendar widget appears. From the calendar widget they can choose a date from the date of the purchase out to 1 year and the time they wish to have their package delivered. The available options are from 9am to 5pm. This information is all echoed back to the user on the console so it can be verified. Once the user picks their delivery date and time, a receipt is generated as a text file called: “receipt.txt” which has a complete summary of their order. This includes all personal information, items, and totals, including taxes, shipping, etc., and the details of their delivery times and dates.

- What are the limitations of the program?

Some of the limitations of the program are the limitations of the items available to the customer, the calculation of the correct tax rate and the logistical factors involved in having the merchandise delivered in a realistic manner are not considered. Generally speaking, an online store has more than 10 items for sale, although this is possible in some niche markets that sell rare or boutique style bespoke items. For this to be more realistic, a catalog of items should be available to the user from different manufacturers, brands, and price points. As I found out calculating the correct tax rate and shipping is an involved process, contingent on the customers location. For example, the state, county, and city where the customer lives all factor in to how the tax rate is calculated correctly. Lastly getting items delivered in a reasonable manner and for a reasonable rate also involves a lot of logistical planning which would affect the customer experience. Realistically if the customer is located far from the business warehouse then shipping times and rates go up exponentially. I am not sure how businesses calculate shipping but I think larger companies like Amazon are logistics companies so they can do it cheaper while other companies must rely on third party delivery

companies like UPS or FedEx. Lastly most users are used to shopping with some type of GUI integration as the store front so in that regard, my program is limited.

- Provide recommendation on improving the limitations of the program.

To improve the limitations of the program, I would suggest integrating a complete and interactive GUI for interacting with the user. Instead of choosing numbers from a menu having a photo and button to add items to the user's cart would be more realistic. Implementing a database would also make managing all the different items for sale easier and scalable. As far calculating the correct tax rate based on state, county, and city, I would recommend implementing some type of web-scrapping capability to retrieve the correct tax rate based on the user's location. The same could be used for getting the correct shipping rate and shipping time. It would not be feasible to have same day or next-day shipping without a substantial increase in the shipping rate and without considering the location of the customer.