

Walter Matthews

COMP 2710

Alvin Lim

November 29, 2017

## **Analysis**

### Use cases:

This program is designed to control a cash register system and process different types of sales. 8 options are present when the system is started. The first 6 being what kind of sale the user would like to make. The first 3 pertain to sales of a “regular” customer. The customer can pay with cash, check, or credit card. Cash will calculate the amount received and if any change should be returned to the customer. A check requires the user to enter the name of the customer and their driver’s license number. A credit card purchase requires card number, expiration date, and the cardholder’s name. The system will process the sale giving a sub-total, then adds sales tax and gives a final total. This system also processes sales for Contractors. These customers will receive a 15% discount on the subtotal. They also have the same payment options as a regular customer. A contractor must provide and contractor’s ID to receive this discount. Option 7 will allow the user to print all the sales made listing an itemized list, total amount, and the type of each sale. Option 8 is an exit from the system.

## **Classes**

### **Sales**

Base class for the derived classes. Stores information such as items and their price, sales type, tax amount, total amount and payment information. This also contains 3 virtual functions that process appropriate sale type

#### **RegularSale**

Define the overriding function process\_sale. This will calculate subtotal and add taxes to give you a final total.

#### **ContractorSale**

Define the overriding function process\_sale. This will first calculate the subtotal then give it a 15% discount. It then will add taxes and give a final total.

#### **RegularCash**

Derived from RegularSale class. It defines the process\_payment function and calculates cash received and change returned to customer. Prints this information

#### **ContractorCash**

Derived from ContractorSale class. It defines the process\_payment function and calculates cash received and change returned to customer. Prints this information

#### **RegularCheck**

Derived from RegularSale class. Contains variables for name and driver's license number. Defines process\_payment to enter and store name and DL number. It prints this information.

### **ContractorCheck**

Derived from ContractorSale class. Contains variables for name and driver's license number. Defines process\_payment to enter and store name and DL number. It prints this information.

### **RegularCreditCard**

Derived from RegularSale class. Contains variables for name, credit card number, and exp date. Defines process\_payment to enter and store variable information and allows for printing of this information. Won't print whole credit card number, last 4 digits are hidden.

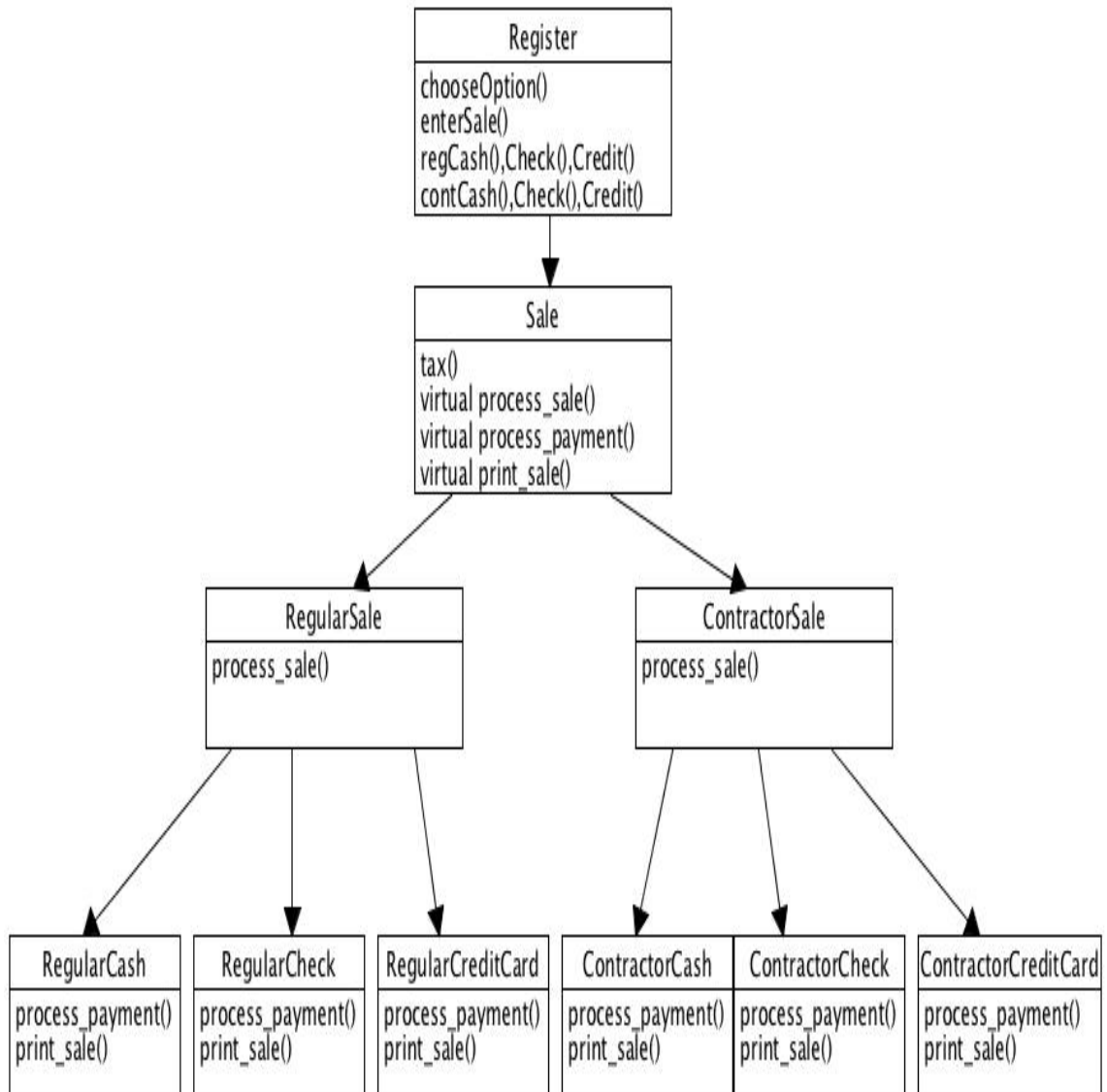
### **ContractorCreditCard**

Derived from ContractorSale class. Contains variables for name, credit card number, and exp date. Defines process\_payment to enter and store variable information and allows for printing of this information. Won't print whole credit card number, last 4 digits are hidden.

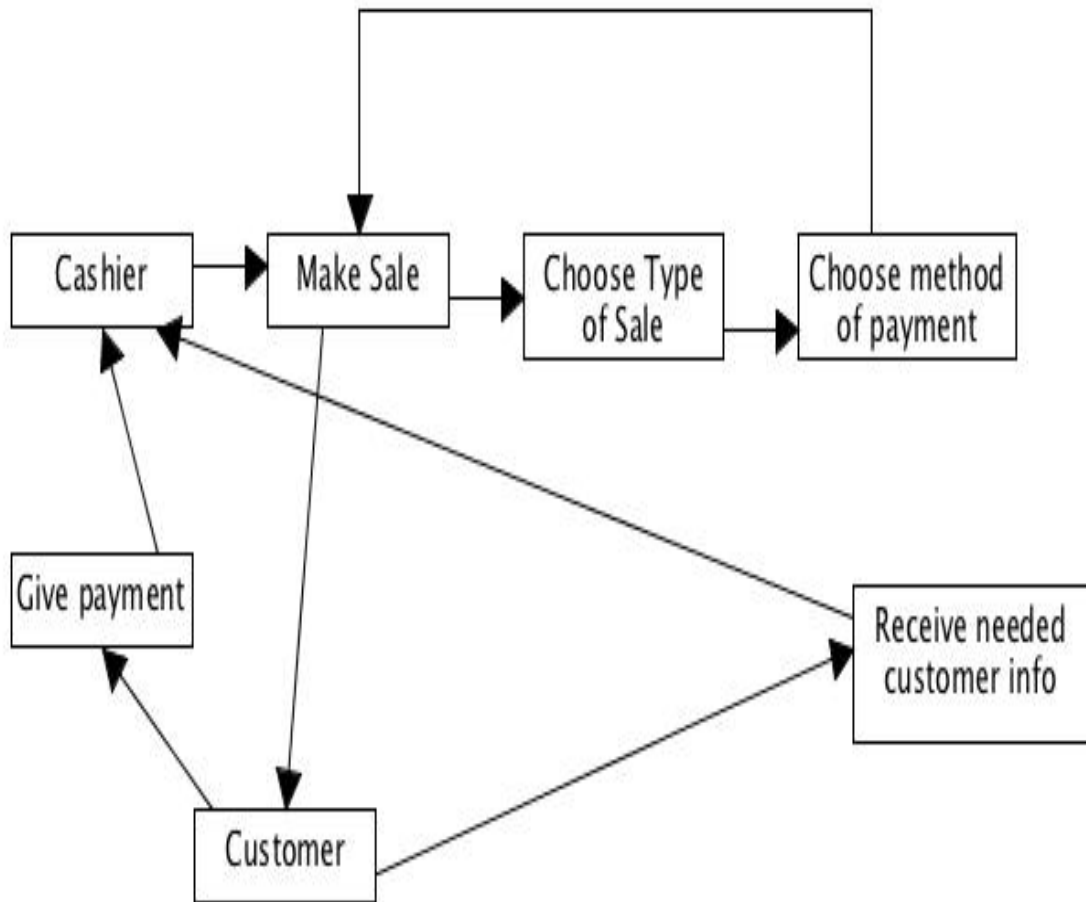
### **Register**

This class allows you to use the actual register. Gives you option to choose an option. Will accept input from user to enter item names and prices. It also defines the action of all options available.

## Class Diagram



## Data Flow Diagram



## Testing

### Register

chooseOption(): only accepts numbers 1-8 or error thrown

enterSale(): only accepts words as an item.

Only accepts dollar amount in XX.XX format

### Sale

tax(): always an additional 8% on subtotal

\$0.00 if subtotal is 0.

### RegularSale

process\_sale(): corresponds with the options for regular customers only.

Prints the correct sale type and sub-total depending on payment type.

### ContractorSale

Process\_sale(): corresponds with the options for contractors only.

Takes subtotal and gives 15% discount everytime.

Prints correct sale type and total depending on payment

### RegularCash

Process\_payment(): Calculates amount received and what change to return.

Print\_sale(): Sale banner correctly printed

Correct number of items listed

Sale type is regular

Payment type is cash

Prints correct amount received and change.

#### RegularCheck

Process\_payment(): accepts name and drivers license number.

Stores it in member variables.

Print\_sale(): Sale banner correctly printed

Correct number of items listed

Sale type is regular

Payment type is check

Prints customer name and DL #.

#### RegularCreditCard

Process\_payment(): accepts name, credit card number, and exp date and stores it in member variables.

Print\_sale(): Sale banner correctly printed

Correct number of items listed

Sale type is regular

Payment type is credit.

Prints customer name, CC# and exp date.

#### ContractorCash

Process\_payment(): Calculates amount received and what change to return.

Print\_sale(): Sale banner correctly printed

Correct number of items listed  
Sale type is contractor  
Payment type is cash  
Prints correct amount received and change.

#### ContractorCheck

Process\_payment(): accepts name, credit card number, and exp date and stores it in member variables.

Print\_sale(): Sale banner correctly printed

Correct number of items listed

Sale type is contractor.

Payment type is credit.

Prints customer name, CC# and exp date.

#### ContractorCreditCard

Process\_payment(): accepts name, credit card number, and exp date and stores it in member variables.

Print\_sale(): Sale banner correctly printed

Correct number of items listed

Sale type is contractor.

Payment type is credit.

Prints customer name, CC# and exp date.