Assignment Report

This report consists of requirements of the system in the form of use cases. UML diagrams including class diagram, sequence diagram and use case diagram of the specified system, and short reflection on the importance of requirement design and analysis, UML diagram and Object-oriented programming with interfaces and polymorphism.

# Use Cases:

The system that we are required to build caters the task of providing tier-based discount to the members on their purchases.

## Create a member:

The member needs to first create their profile in the system. This enquires their name and the tier of membership they want to choose. The company provides membership comprising of a regular member, bronze member, silver member, gold member and platinum member.

## Make a purchase:

Once users have registered in their system. Their profile is stored. Once member have purchased a product. The system displays them their discounted price with respect to their membership. Discounts are given according to following policy:

|  |  |
| --- | --- |
| Membership | Discount |
| member | 6% |
| Bronze member | 10% |
| silver member | 13.5% |
| Gold member | 19% |
| Platinum member | 25% |

## Display a summary:

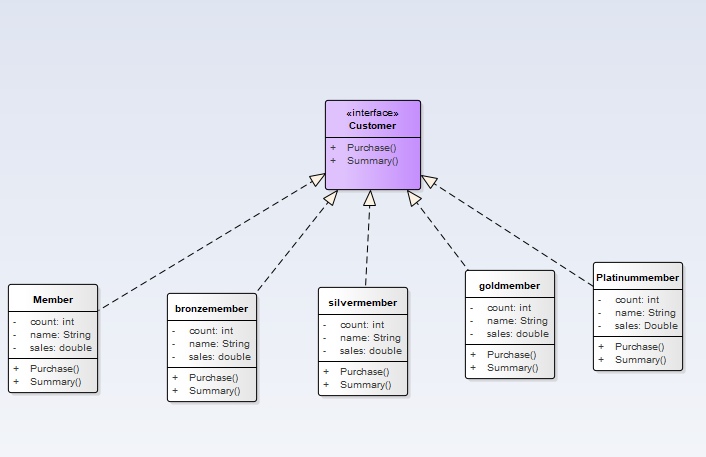
This option provides a detailed summary of previous total sales and payment received and sales and amount received in each tier as well. Total discount given to each membership tier is also shown.

## Exit the system:

This option ends the processing of system.

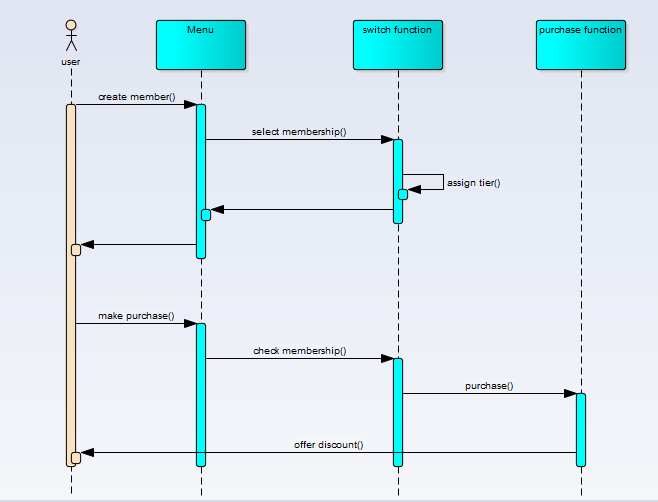
# UML Diagrams:

## Class Diagram:



This above figure represents a single interface called Customer. This interface contains two functions mainly purchase() and Summary().All the classes below implement the functions of this interface but every function has its own definition of the same function. Every class has three attributes, name contains the name of member which subscribe for membership. count attribute information for total sales in each tier. Sales contains the amount received in each tier. Purchase() function in different classes offers different discounts according to tier.Summary() function displays summary of each tier.

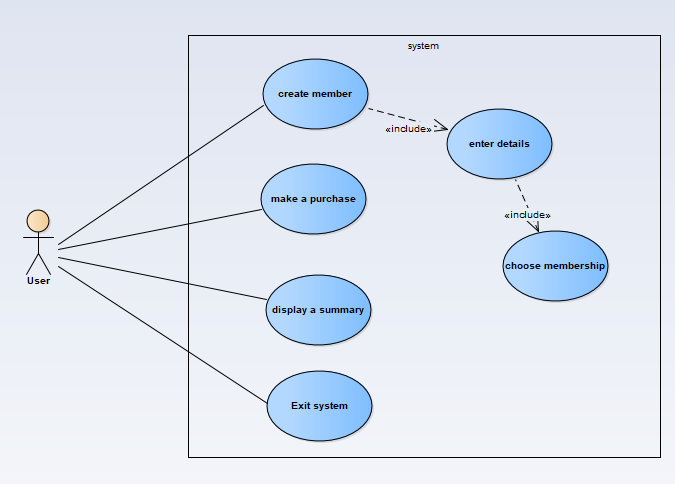
## Sequence Diagram:



The above figure shows sequence of operations for purchasing from the system. In order to make purchase. Use must register as a member in the system and choose the membership. Once the user is registered, he can make purchase. After the user selects purchase option, he/she enter retail price to the system. The system then checks user`s membership tier and offers discount accordingly.

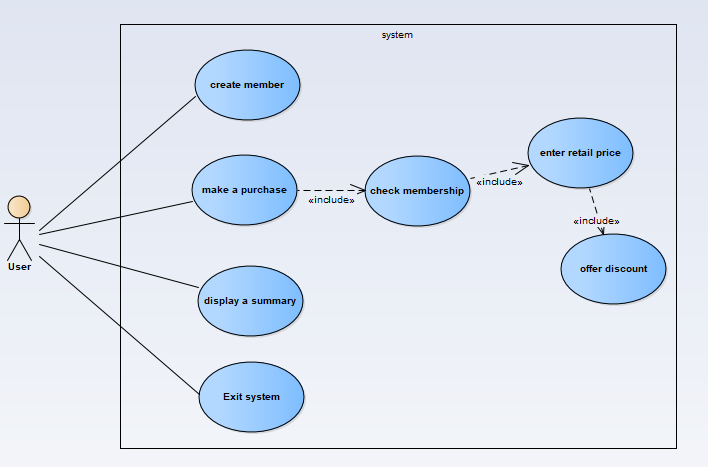
## Use Case Diagram

### Create member:



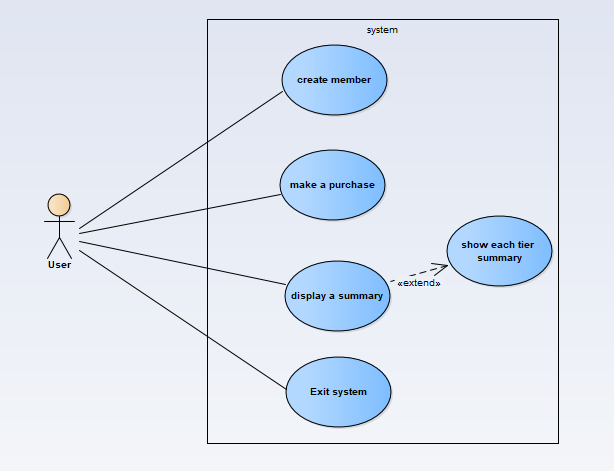
In order to create member, The system further uses a function which takes details from the member and offers him different memberships, User has to choose among five different membership tier.

### Make Purchase:



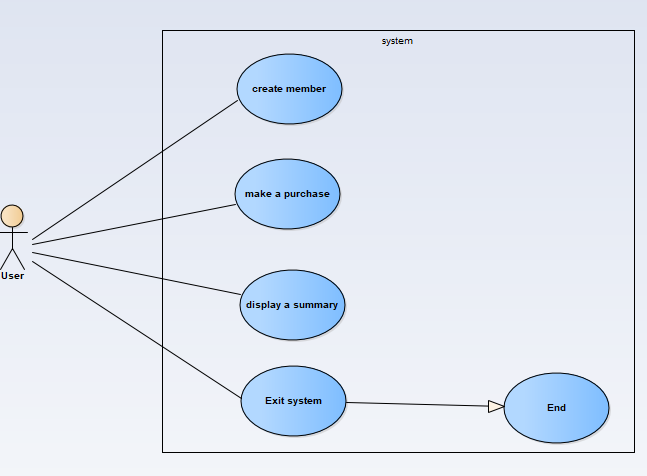
When user selects for purchasing an item, The system checks the user record to identify their membership tier. The system than asks the user to enter retail price, the individual purchase functions calculate the discount and deduct the amount from the retail price.

### Display a summary



When the user selects summary option, the system calls the summary function, which in turn calls the summary() function of each membership class. Total sales, amount received, and discounts offered in each tier are than shown to user. After that, total sales and amount received in all tiers are summed up and presented.

### Exit system:



This option simply end the programs and deletes the user data.

# Discussion:

Requirement design and analysis is a part of requirement engineering. It is very essential part of software development life cycle. No one can be a good software engineer without the knowledge of software design, planning, analysis and requirement engineering. The one who only code is a developer only. Software project management is very important in businesses. Software companies that produce enterprise level software always work by starting with requirement gathering and design of product. Throughout the history, many projects which were heavily funded failed to build because of lack of management and plan.

To cater the need of system design, UML diagrams are used to cover structural and behavior model of the system. These diagrams include use case diagrams, sequence diagram and class diagram etc. These all diagrams should be made before the development phase in order to avoid failure in future.

There are two ways of writing program. One focuses on what the program does and is called process-oriented model. The other approach is object-oriented programming, it is a way of doing programming considering every essential part of the system as an object. For example, if we have to write a program for atm machine. We will treat the atm machine as an object. Now, the object may contain it own attributes and function. In this case, Balance, amount will be attributes, meanwhile, taking cash out is a function of atm. In object-oriented programming, the objects are represented as classes. Polymorphism is a feature that allows one interface for a general class of actions. The subclasses use the same function but have their own different implementation. This approach helps to add more classes in the code without editing in middle of code.