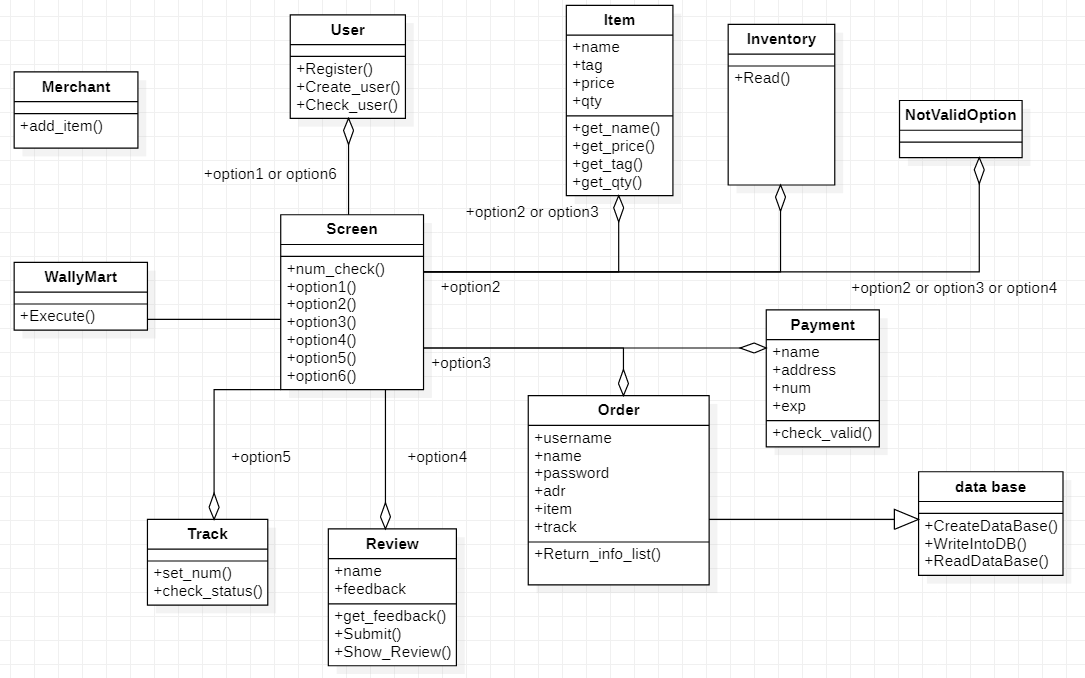
**UCSC extension: Object-Oriented Analysis and Design**

**Project X**

**Ting-Tzu (Claire) Chiu**

**UML diagram:**

****

**Architecture layer diagram:**

****

**Architecture flow diagram:**

****

**Sequence diagram:**

****

**Code documentation:**

This is a program written in python to solve WallyMart problem. It contains 12 python files, 3 data base and one excel sheet.

**Data Base**

**User:**

Use to store username and password. If user try to login it will link to this database and check whether he/she is a valid user.

**Order:**

Use to store username, name on card, address, items, track number, date and status that a successful purchase made. If all info correct during check out it will connect to this data base and store all info into it.

Review:

Use to store username, review text and date. If user write a review (must login) it will connect to this data base and store info into it.

Into it.

**Excel sheet**

**Inventory\_list:**

This is a list that admin can add more item for user to select.

**Python Code**

**Main:**

Admin page. You can select if you are a user or a admin. In this page you have 3 selection as

below

1. Customer (call Wally Mart class)
2. Merchant (call Merchant class)
3. Quit

**Merchant:**

This class provide a method to add item into Inventory\_list, If add item is not in current category

it will create a new category (new tab) and update Inventory\_list.

**WallyMart:**

This class provide a method called “Execute”. It provide different selection that you can make as

a user. In this class it will call Screen class to provide function for 1.Login, 2.Shopping, 3.Cart,

4.Reviews, 5.Check status, 6.Register, 7.Goodbye!

**Screen:**

In this module will provide 2 class. One is called “NotValidOption” inherit from Exception class. It

will be use to check if user input an invalid selection.

The other class called “Screen” and it will provide below methods:

1. num\_check : This check if the input is digit only
2. option1: Use to do “Login” action. Will use “User” class method “Check\_user” to obtain username and password list from data base then check if this is a valid login
3. option2: Use to do “Shopping” action. Will use “Inventory” class method “Read” to dump Inventory\_list category and based on user’s selection will show variety choice under this category. User can select item and quantity as many times as you wish. Will call “Item” class and append these info into a buy\_list .After finish shopping just select “check out” and will go back to main page.
4. option3: Use to do “Cart” action. User can either check if you have items in your current shopping cart or check out. If user want to check out will first check if login or not (call option1). After user login will call “Payment” class method and dump purchase info (e.g name on card, shipping address, credit card number, exp date) into it. Will call Payment” class method “check\_valid” to check if credit number is valid or not (currently check if it is only 5 digits). If all the info is correct will generate a tracking number and call “Order” class method “WriteIntoDB” to store this purchase into Order database then you will go back to main page.
5. option4: Use to do “Reviews” action. User can either write a review or check recent review.If user want to check out recent review user can select how many review he/she would like to read and call “Review” class method “Show\_Review”. If user want to write a review will firstly check if user already login or not. If not login will call option1 to login first. If user already login he/she can write a text review. Once finish writing will call “Review” class method “Submit” to store into Review data base.
6. option5: Use to do “Check status” action. User would get a tracking when a successful purchase made. User can input this info and will call “Track” class method “check\_status” it will return current status for your purchase.
7. Option6: Use to do “Register” action. Will use “User” class method “Register” to write into current User data base. If username already exist will not allowed to write in.

**Inventory:**

This class provide a method called “Read”. Will read Inventory\_lsit.xls and return a list with

(item, price, tag)

**Item:**

This class use to store info from purchase item. It will store name, tag, price and qty

**Order:**

This class provide a method called “Return\_info\_list” will return a list with this content

[username, name, adr, item, track, time, status]

**Payment:**

This class provide a method called “check\_valid” it will check if the input only contain5 digits.

**Review:**

This class provide three methods called “get\_feedback”, ”Submit” and ”Show\_Review”

get\_feedback : Will return input feedback

Submit: Will connect to Review data base and store username , review and time into it

Show\_Review: Will let user to input a number and will connect to Review data base to read out

numbers of review start from the latest time

**Track:**

This class provide two methods called “set\_num” and “check\_status”

set\_num : will return the string you input

check\_status: take a string input and connect to Order data base to find track column if this is a

valid package

**User:**

This class provide three methods called “Register”, “Create\_user” and “Check\_user”

Register: Will connect to User data base and check if input username exist. If yes will ask to use

a new name otherwise will write username and password into data base

Create\_user: Use to create a data base.

Check\_user: Will connect to User data base and check if input info match with current data base

info

**data\_base:**

This class provide three methods “CreateDataBase”, “WriteIntoDB” and “ReadDataBase”

CreateDataBase : Will create a data base named “Order”

WriteIntoDB: Will connect to “Order” data base and write info into it

ReadDataBase: Will connect to “Order” data base and read out sorted (latest first) info from it

**Use case:**

Brief:

A user can use WallyMart program to become a member. A user can shop variety of items based on different categories. A user can write a review based on his/her shopping experience. A user can use tracking number got from a successful purchase to check delivery status.

Full:

|  |  |
| --- | --- |
| Name | UCSC-OOAD-Final WallyMart in python code |
| Scope | Engineering system Level |
| Level | System Goal Level |
| Primiary Actor | the user for shopping online |
| Stakeholder | WallyMart |
| Preconditions | This must come with WallyMart's official site |
| Postconditions | User, Order data base will keep updating |
| Main Success Scenario | see Main Success Scenario in below |
| Extensions | see Extensions in below |

Main Success Scenario:

1. While running Main program you can select 1 to start shopping
2. User select 1 can login into server once login it will go back to Main page
3. User select 2 can start shopping depends on selection
   1. User select an item can decide how many quantity he/she would like
   2. User can select “check out” and will go back to main page
4. User can select 3 to check items in current cart
   1. User can select 1(Review Order) and it will show current item in the cart then go back
   2. User can select 2(Check out) to check out.
5. User need to input valid info (card number, exp date, address, name etc)
6. Once a valid info get in will send you tracking number and go back to main page
7. User can select 4 to view recent reviews or write review
   1. User select 1(Write a review) can write review and send to data base then go back
   2. User select 2(Check most recent review) can view latest review then go back
8. User can select 5 to input your tracking number to check status then go back
9. User can select 6 to register a new account, it will be saved in data base
10. User select 7 will go back to identity selection page

Extensions

1a1 If user input not 1,2,3 it will keep asking till you select from 1,2,3

2a1 If user input not 1~7 it will keep asking till you select from 1~7

3a1 If user select an out of range item will jump back to previous page

4ab1 If cart is empty will jump back to previous page

4ab2 If cart is not empty but user doesn’t login will ask user to login

5a If credit card number is invalid will keep asking till valid card number acquire

9a If username already exist will ask user to create another one