Vincent Nguyen

nvui@bu.edu

System Design For BestPurchase app

Table of Contents

[Introduction 1](#_Toc128005611)

[5.1 Class Model 2](#_Toc128005612)

[5.2 Packages 3](#_Toc128005613)

[5.3 Logical Data Flow Diagram 4](#_Toc128005614)

[5.4 Physical Data Flow Diagram 6](#_Toc128005615)

[Conclusion 6](#_Toc128005616)

[References 7](#_Toc128005617)

[Evaluation 8](#_Toc128005618)

Table of Figures

[Figure 1: Example of Mobile retail app 1](#_Toc128005623)

[Figure 2: Updated class model 2](#_Toc128005624)

[Figure 3: Class model with packages 3](#_Toc128005625)

[Figure 4: Logical Data Flow Diagram 5](#_Toc128005626)

[Figure 5: Physical Data Flow Diagram 6](#_Toc128005627)

# Introduction

The BestPurchase app’s scope is to provide a platform for customers to shop for groceries from difference stores in one go via phone or website. The BestPurchase allows users to explore products, choose items, place orders, and have orders deliver to their home. The BestPurchase also provide scheduling delivery services.

In this assignment, previous class model is updated with additional classes to provide core packages for design system to get low level of coupling and high level of cohesion. There are 3 core packages in the class diagram for BestPurchase use case. Logical data flow diagram is designed to show how data process by the system in term of input and output. Physical data flow diagram is also designed to show the implementation details and explain how the system will work.

Graphical user interface

Description automatically generated

Figure 1: Example of Mobile retail app

# Class Model

The following class model has been derived and showed the high level of classes which are necessary for the customer to perform online shopping and payment in the use case from the previous assignment. The class model has been updated with several changes from the previous class model in order to incorporate the packages in the model. Firstly, some classes were added to the model such as LoyalCustomer, BuyingCustomer, Electronic, Grocery. The relationship between those classes is also defined. Secondly, some relationships have been changed and multiplicities were added in those relationships. Third, the Driver class was deleted.

* The changes are highlighted with orange color.
* The additions are highlighted with blue color.

Diagram, schematic

Description automatically generated

Figure 2: Updated class model

# Packages

Diagram, schematic

Description automatically generated

Figure 3: Class model with packages

In the update class diagram. There are 3 packages below:

* **Package name**: Customer
* **Design goals for this package**: Customer package focuses on sufficiency, flexibility, reusability, and goals.
* **Design tradeoffs for this package**: Customer package provides sufficiency to the system because it allows customer to interact with the system through <<Main> BestPurchase class, and it requires the customer to create account via this class. It also allows system to meet flexibility and reusability because new type of customer can be added to the supper class Customer due to inheritance relationship between them. There is high level of coupling within the Customer package because there is inheritance relationship between supper class Customer and sub classes such as LoyalCustomer and BuyingCustomer (This is a type of customer who just buy the thing they need; they will not shop around). Customer package and other is low degree of coupling because there is dependency relationship between Customer package and <<Main> BestPurchase (It won’t be boxed as package due to having only one class). Therefore, it is high cohesion.
* **Package name**: Shop Products
* **Design goals for this package**: Shop Product package focuses on sufficiency, reusability and flexibility and efficiency.
* **Design tradeoffs for this package**: One of potential tradeoffs for Shop Product package is flexibility and maintainability. The Shop Products package should be flexible to adjust different changes in the products type, deals. However, if there are too much flexibility can make too difficult to maintain. Another potential tradeoff for Shop Product package is sufficiency and robust. The Shop Product package should be sufficiency for all of requirements but if there are too many requirements in the big system, resulting harder to handle unusual conditions. The Shop Products package has high level of coupling within this package because there are several relationships within the package such as composition relationship, aggregation, and inheritance relationship. The Shop Product package has low degree of with other packages due to aggregation relationship between them. Therefore, it has high cohesion.
* **Package name**: Purchase
* **Design goals for this package**: Purchase package focuses on sufficiency, reusability, and flexibility
* **Design tradeoffs for this package**: One of potential tradeoffs for Purchase package is flexibility, reusability, and usability. The Purchase should be flexibility and reusability but with large online system, there will be too much flexibility and reusability, it could lead to low usability, poor user experience. The Purchase package has high level of coupling within the package due to composition and inheritance relationship between classes, but Purchase package has aggregation relationship with other package, therefore it has low level of coupling and high level of cohesion.

# Logical Data Flow Diagram

The below logical data flow diagram illustrates how BestPurchase app operates and its core functionality of its as described in the use case from previous assignment. It describes the processes that take place and the data which are required and produced by each process. The diagram below consists of a series of bubbles which represents processes. The arrow lines represent the flow of data (input and output data). The yellow square presents external agent (the customer in this use case). The green rectangles represent the data stores.

Diagram

Description automatically generated

Figure 4: Logical Data Flow Diagram

# Physical Data Flow Diagram

Diagram

Description automatically generated

Figure 5: Physical Data Flow Diagram

In this use case, The customer already login in the BestPurchase app.

1. The customer search product through text input into the BesPurchase Phone Mobile App.
2. The BestPurchase Phone Mobile App send product as text to BestPurchase Server.
3. The BestPurchase Server performs Product query search against the Product data store.
4. The Product Infor is sent back to the customer through the BestPurchase Phone Mobile APP – it will be displayed on the Mobile Device Display.
5. The customer initiate payment by input Card Number text into the BestPurchase Phone Mobile APP, the application running on the BestPurchase Phone Mobile APP will will query Account Infor by sending Account Number to the Card Provider Server, Card Provider Server will query Account Number from the Accounts data store and send back Account Infor to BestPurchase Phone Mobile APP through Card Provider Server.
6. The Mobile Device Display will display the transaction message (Transaction Successful Message or Transaction Failed Message) to the Customer.

In this design, Display and Mobile APP are located on the mobile.

# Conclusion

This assignment has been designed based on previous assignments which is assignment 3 and 4. The class model has been updated with more classes to define the packages. There are three packages has been defined and a few classes has not defined as package because it is the only one class itself. The logical flow diagram is designed to describe the business events that take place and the output data of those events and physical flow diagram also is designed to show how a system is implemented.

# References

[1] Dennis, A., Wixom, B. H., & Tegarden, D. (2015). Systems Analysis & Design an Object-Oriented Approach with UML (5th ed.). Wiley.

[2] <https://lucid.app>

[3] Williams, Dawson Lee. (2017). Module 0. Executive Writing Video.

[4] Williams, Dawson Lee. Week 5 Class Lecture. Boston University MET CS 682 O1

[5] Arakelian, A. Week 5 Assignment Preview. Boston University MET CS 682 O1.

[6] Logical diagram. <https://www.youtube.com/watch?v=ztlQvpS4QHk&t=223s&ab_channel=JordanBarlow>

[7] Data Flow Diagram. <https://www.youtube.com/watch?v=6VGTvgaJllM&ab_channel=SmartDraw>

[8] Scott, Cromar. Week 5 Open Office. Boston University MET CS 682 O1.

[9] Williams, Dawson Lee. Week 5 Open Office. Boston University MET CS 682 O1.