Seminar 1

Object-Oriented Design, IV1350

Amanda Koinberg, amakoi@kth.se

19/04/2023

Contents

1: Introduction

This task aims to design a program that can represent a store sale process by using Astah diagrams. These diagrams will represent a programming flow of a sale process. The task is explained in detail on the course's canvas page. This report will show a possible solution to the assignment and discuss the solution.

I worked alone while doing this assignment.

2: Method

The methods of creating design diagrams comes from the course literature, which wants us to create a MVC (model-view-controller) pattern, a common practice when designing object-oriented programs. We create a package diagram with layers and each layer should contain at least one class (of the layer itself), but if the program is large then several controllers, models and views may be necessary. For this task, only one layer class of each layer should be enough. I used the guides in the literature as well as general guides on MVC that I found online for further explanations and examples. MVC is a well-used practice and it made it easier to understand how I should set up the program.

3: Result

Five diagrams were created describing the flows of the sale process. In figure 3.1, we can see the startup process where all the main classes and the system is created. It is the main components of the system.

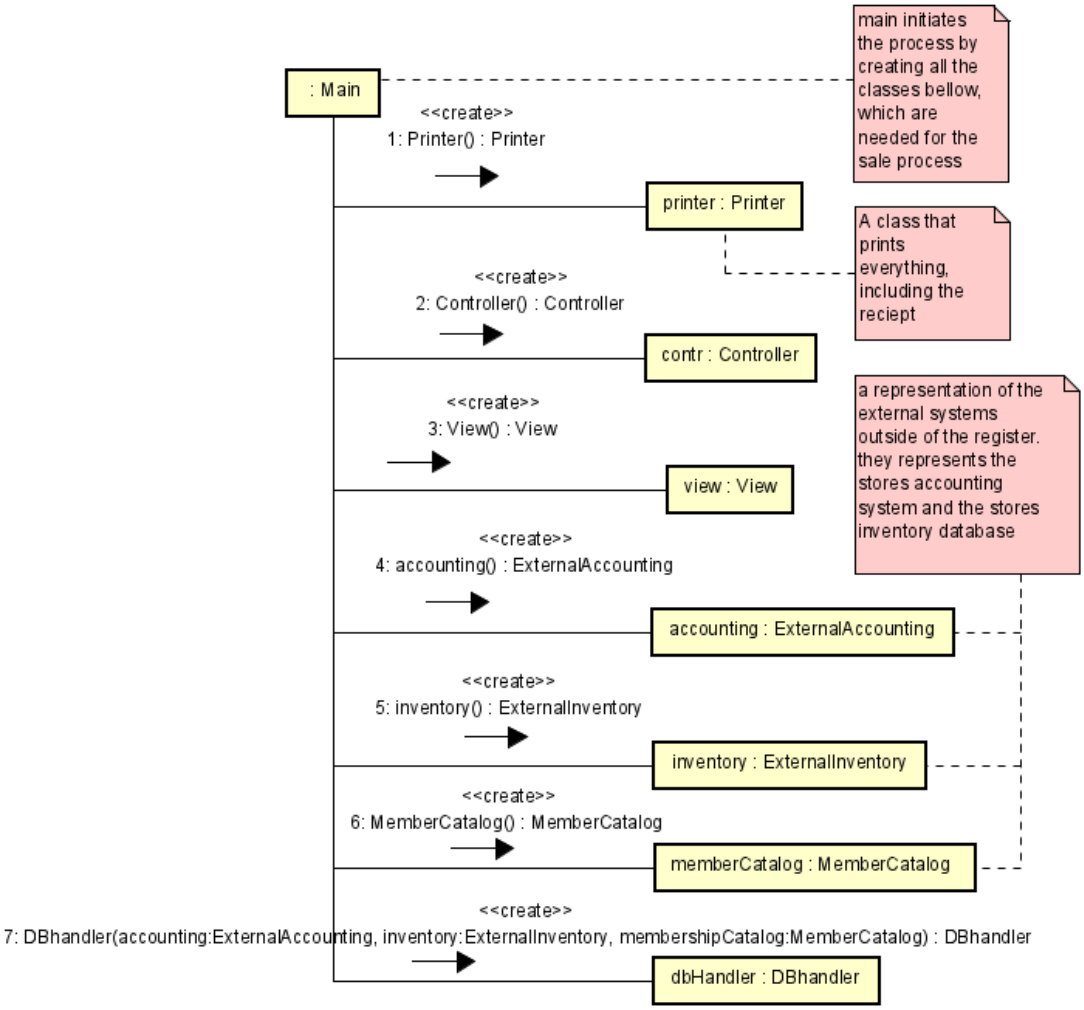

Figure 3.1 Startup

In figure 3.2, the process of starting a new sale is showed. This process is initiated before the cashier starts scanning items. The necessary classes are created so the sale can take place.

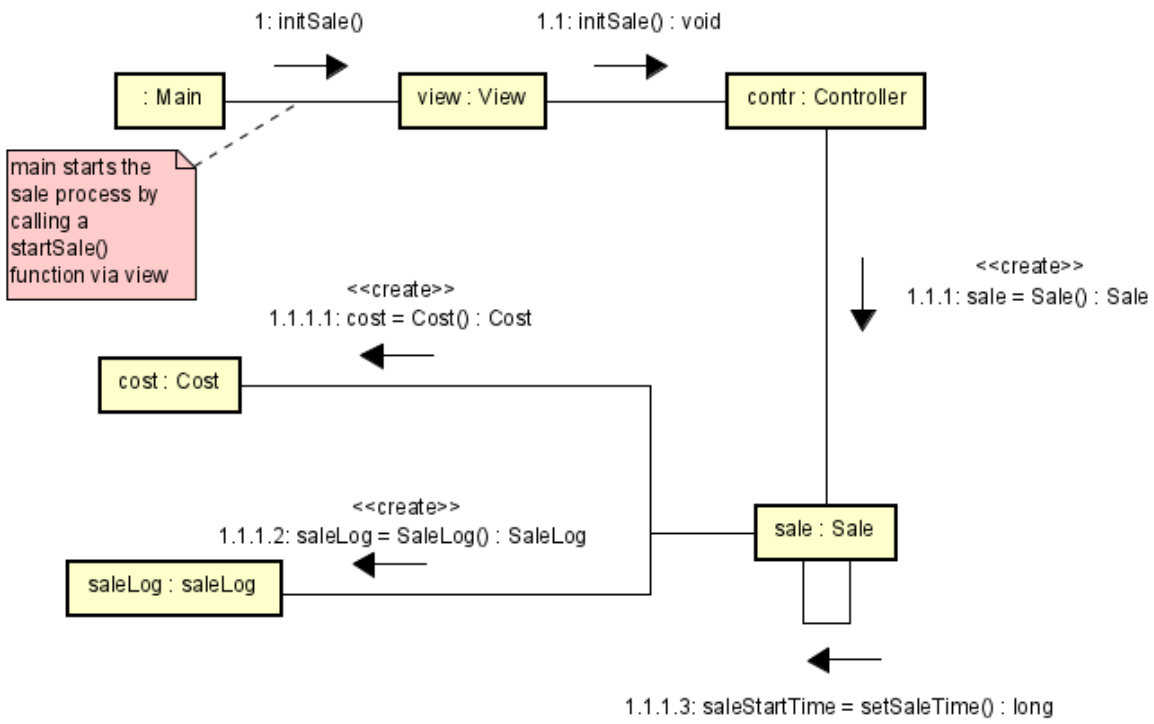

Figure 3.2 Start Sale

Figure 3.3 shows the process of when the cashier scans items and adds them to the sale log. Each scanned item updates the sale log and adds the item cost to the total cost amount.

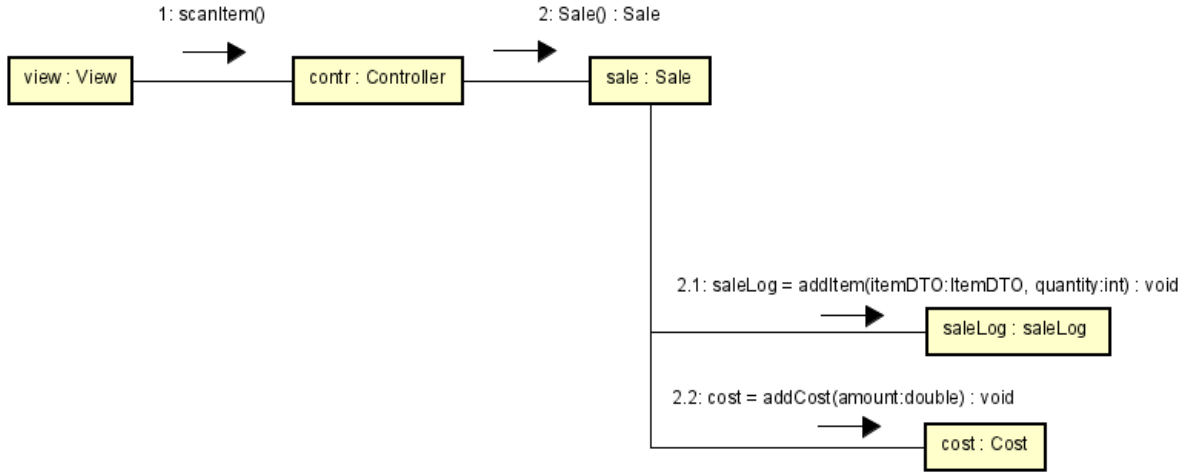

Figure 3.3 Add item

When the cashier is done scanning, then the cashier may ask the costumer if they want membership discounts. In that case, if yes, then the system will check the membership database to find the member in the member catalog and add the discounts (1.1). See figure 3.4. The system will then finish (weather we used the discount or not) and end the sale. We calculate the total cost and enters payment methods.

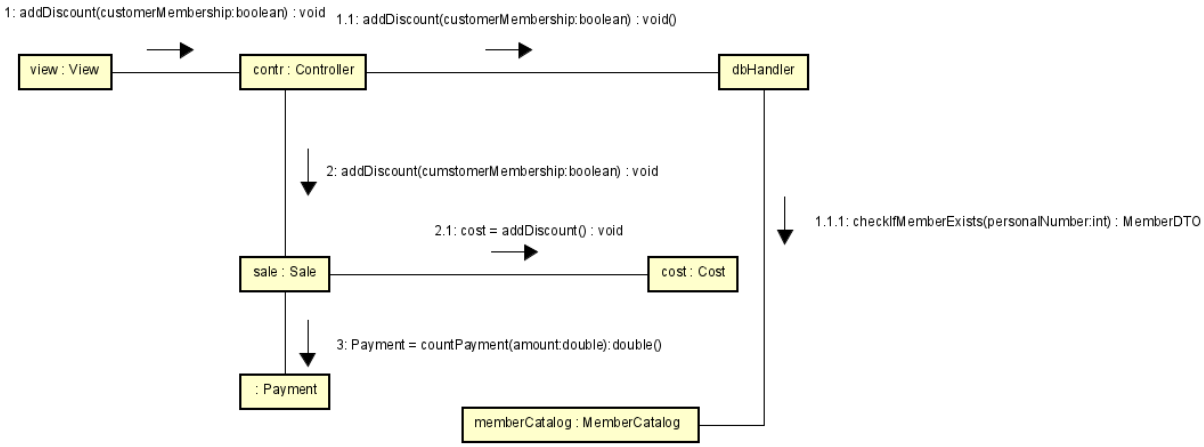

Figure 3.4 Add discount and pay

Figure 3.5 shows the process after a sale is done. It needs to be closed. The cash register will be updated, external databases will also be updated. The customer will get a printed receipt.

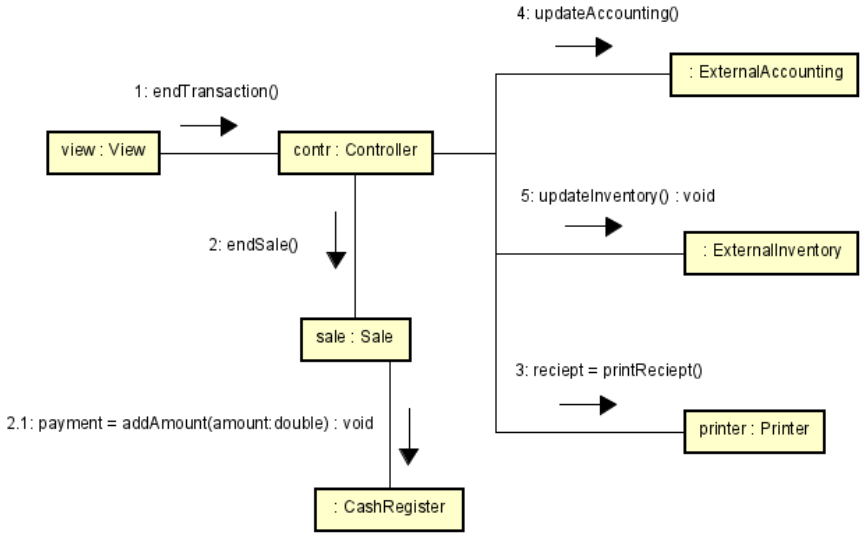

Figure 3.5 update database

The final package diagram can be seen in figure 3.6.

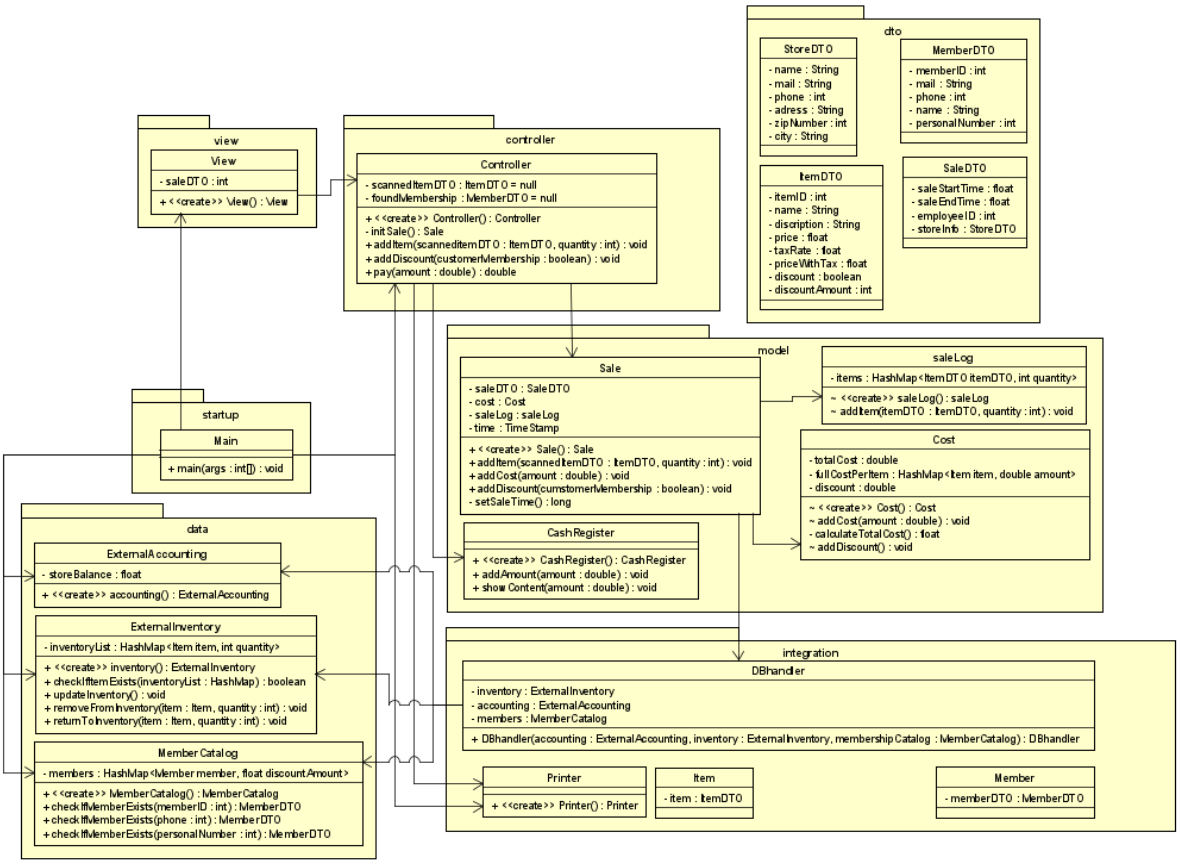

Figure 3.6 package diagram

4: Discussion

My solution should be easy to understand, it follows a logic red line from start to finish. I use the layer patterns that was mandatory, and the classes do what they should.

The result showed in chapter 3 are following the criteria of the assignment and I have delivered what was asked of me. Although I solved the assignment, there is probably room for improvement, like always. I could spend more time trying perfect it, but a model is not a finished product so I must not stay here.

I thought it was hard to make the flow charts in seminar 1, and it became clearer to me what mistakes I made while working on seminar 2. I assume that when I start on the seminar 3 assignment, then I will realize I made some errors here as well.