

Seminar 2

Object-Oriented Design, IV1350

Nariman Haidar

narimanh@kth.se

22-04-2021

Contents

1 Introduction 3

2 Method 4

3 Result 5

4 Discussion

1 Introduktion:

Programmet är delade i lager baserat på MVC- metoden för en optimerad objekt orienterad design. Designen är kommunikations diagram för huvudklassen MVC och lager mönster. Jag samarbetade tillsammans med Tommy Hanna, Sümeyya Bayram och Mickel Mounirji.

2 Metod:

Jag började med att skapa klassdiagram och skapade paketen view, controller, integration och metod. Sen i andra steget skapade jag ett kommunikation diagram, för varje systemoperation som ny försäljning, registerartiklar, rabatt, betalning. För att visa hur en funktion skulle fungera och för att visa processen och kopplingen mellan objekt använde jag en kommunikation diagram.

För att designa ett systemoperationer med en hög sammanhållning skapade jag nya klassar i lager som varje klass representerar en abstraktion som identifierar klassen. Det sista steget var huvudklassdiagram som visar alla objekt som finns i systemoperationerna. För att visa hur en funktion skulle fungera och för att visa processen och kopplingen mellan objekt använde jag en kommunikation diagram.

3 Resultat

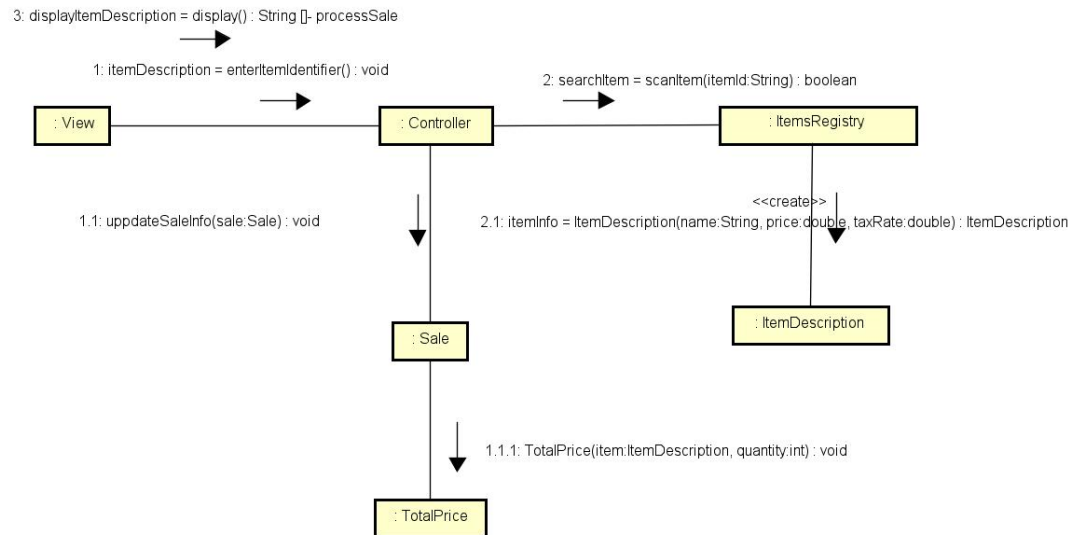
Starta ny försäljning:

Den här diagram är första systemoperationen i sekvensdiagrammet. För att kunna starta en ny försäljning leder till att skapa en ny operation CurrentSale i kontrollerklass i paketkontrollen och sen skapa operationskvittot.

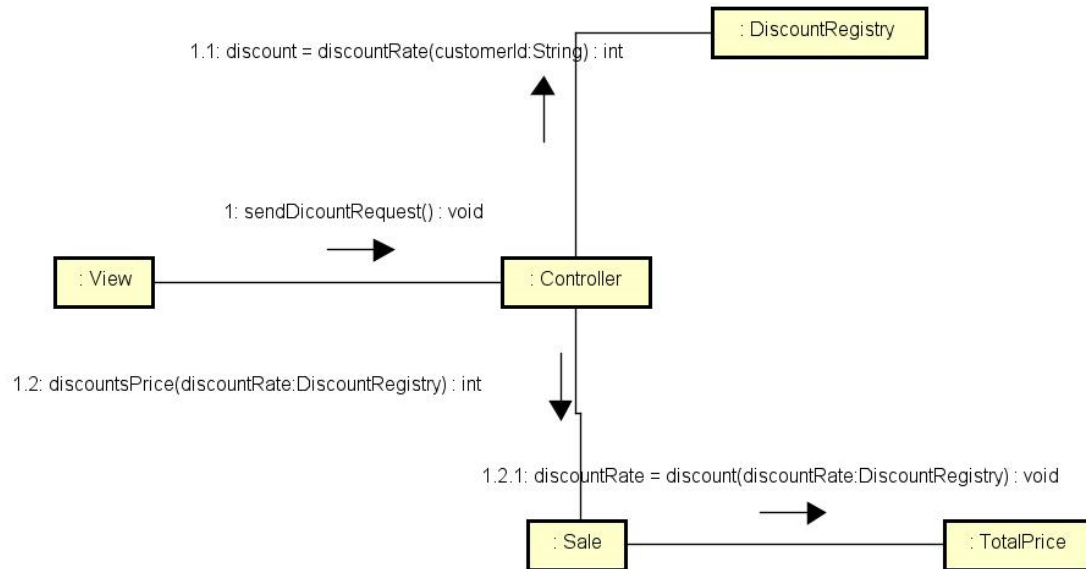


Registrera objektoperation:

I den här systemoperationen för att börja registrera objekten i systemet så ska man göra enterItemIdentifier och få deras beskrivning representerad på skärmen.

**Rabattooperation:**

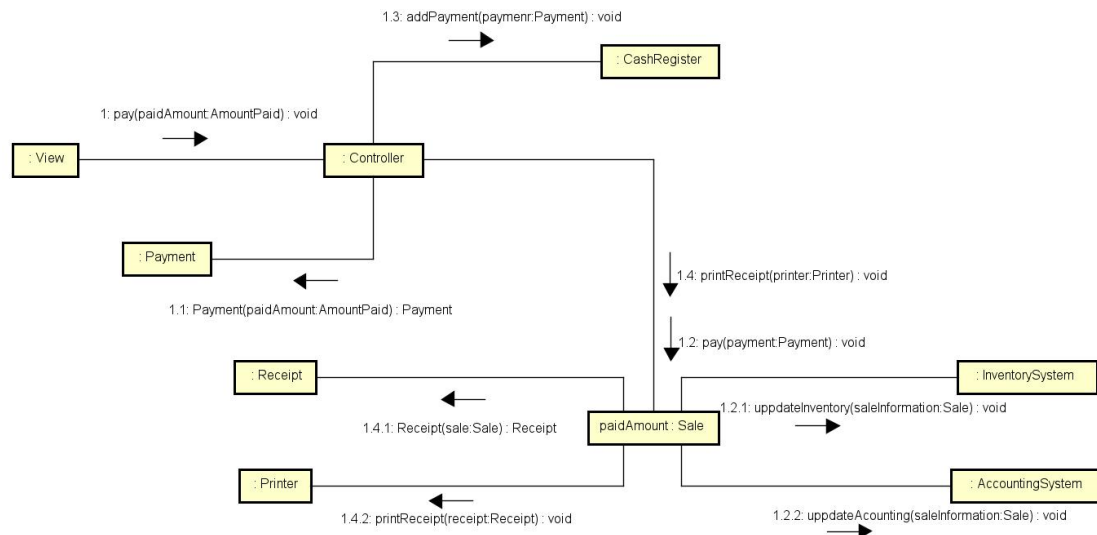
Diskonteringsröntan ska räkna efter att kontrollern registrerar kunden i systemet genom kunden id och veta att kunden har rätt till Diskonteringsröntan. Därefter uppdateras rabattvärdet i den aktuella försäljningen sen totalpris.

**Betalning:**

Den här systemoperationen är sista systemoperationen. Det betalade beloppet läggs direkt till kontanterna

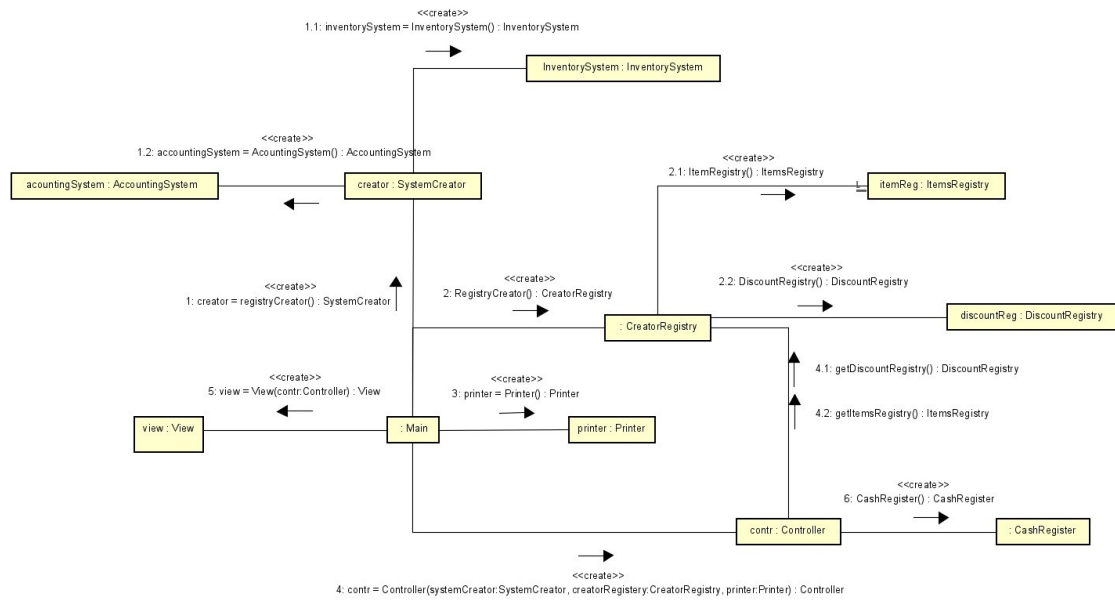
Registrera. Alla information ska vara i kvittot.

För att kunden får kvittot så kommer klassen kvitto att få en beställning med försäljningsinformation och sen detta kvittot ska skicka till skrivaren och komma ut till kunden.



Huvudoperationen:

Här är huvuddiagrammet. I huvudmetoden skapade jag en centrala objekt som inte nämnts i tidigare diagram. Jag försökte att göra lågkoppling.

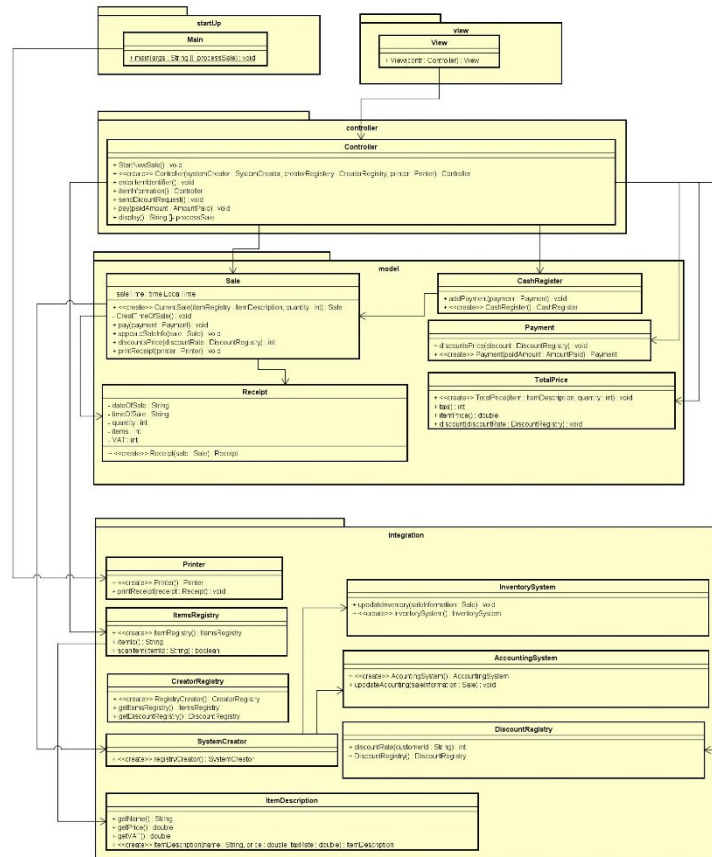


Klass diagram:

Slutligen lade jag till alla klasser som används i interaktionsdiagrammen och beställde dem i lager av

MVC. Sedan baserade jag på interaktionsdiagrammen och gjorde associationerna och riktningen

mellan klasserna. Allt som View använder går genom styrenheten som skickar instruktioner i en rad klasser och ge den en låg koppling.



4 Diskussion:

Den här uppgiften var svårare än det första för att det tog tid att förstå hur man skapar systemoperationerna och hur den kommer att implementeras i diagrammen eller hur koden kommer att se ut. Jag tog hjälp av boken och kollade på föreläsningarna och gjorde övningarna. Jag tycker att det gick ganska bra, de är alla lätta att förstå, namngivningen är tänkt att göra det lättare. Jag gjorde MVC-diagrammet som har alla lager som behövs. Jag fokuserade lite mer på returvärdena och typerna för detta seminarium så det blir tydligare. Klasserna är tydliga och har hög samanhållning i diagrammet och inte så högt så det gör det svårt att koda. När det gäller inkapsling tycker jag att det är rätt.