

Seminarium 1

Objekt-Orienterad Design, IV1350

Daniel Westerlund

daweste@kth.se

2019-03-27

Innehållsförteckning

1	Introduktion	3
2	Metod	4
3	Resultat	5
4	Diskussion	6

1 Introduktion

Seminarium ett gick ut på att bekanta sig med ett UML modellerings hjälpmedel, öva på att skapa en domänmodell samt att öva på att skapa ett systemsekvens diagram. Detta skall lösas genom att skapa en domänmodell utav en detaljhandelsbutik i ett UML program, och därefter skapa ett systemsekvens diagram utav denna domänmodell.

Jag har arbetat själv.

2 Metod

Först gick kravspecifikationen för "Process sale" igenom och substantiven noterades för att utgöra ett underlag till domänmodellens klasser. För att få ett ännu större underlag till domänmodellens klasser gick igenom hur ett köp går till "i verkligheten". Som UML modelleringsprogram valdes [Astah UML 8.1.0/3ac74f Model Version: 38](#). Alla substantiv skapades som klasser i Astah. Nu påbörjas processen att hitta lämpliga associationer mellan klasser samt att avgöra om någon klass bör gå in som ett attribut i en annan klass, eller rent av tas bort från domänmodellen. För att hitta lämpliga associationer mellan klasserna, hittar man klasser som hör ihop med varandra och försöker kombinera ihop dem i en mening med ett verb i mellan. Klasserna som gick ifrån att vara en egen klass till att bli ett attribut, valdes genom att det är klasser som beskriver egenskaper utav en annan klass. Under arbetat av att addera associationer uppstod det i några fall behov utav att addera en/flera attributer till vissa klasser. Detta resulterade i domänmodellen i kap 3.

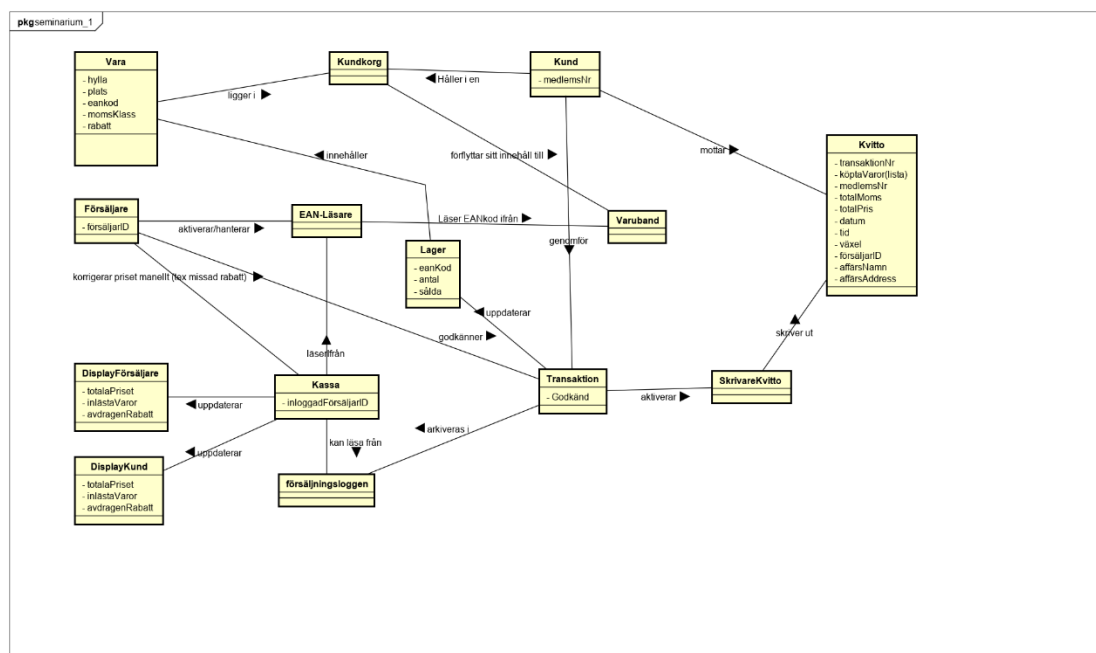
Systemsekvens diagrammet skapades även det i samma Astah program. Genom att ange det olika system som personerna kommer att kommunicera med, de externa system som finns i domänmodellen. Skapa ett flöde av interaktionerna mellan personerna och systemen och i vilken ordning det sker.

3 Resultat

Underlaget för klasser som identifierades ifrån kravspecifikationen var:

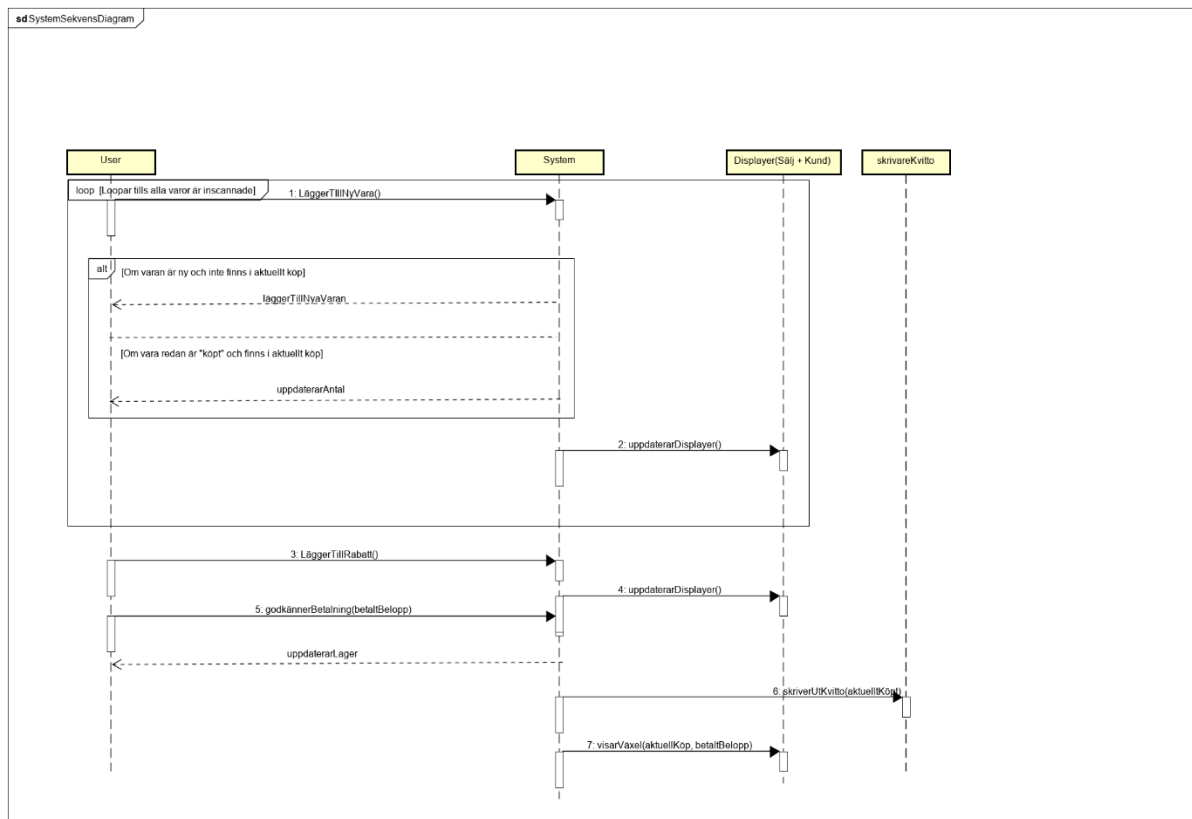
- Kassa
- Säljare
- Kund
- Vara
- Kundkord/Kungdvagn
- Varuband
- Display för säljare
- Display för kund
- Kassa
- Hylla (för varor)
- Transaktion
- Lager
- Skrivare (kvitto)

Figur 3.1 visar resultatet efter att vissa klasser har blivit till attributer samt att det under arbetets gång har upptäckts behov av flera attributer. Samt att associationer mellan klasserna har lags till.



Figur 3.1

Domänmodellen i fig 3.1 resulterade i systemsekvens diagram i fig 3.2.



Figur 3.2

4 Diskussion

Hade en del problem med att det vart "spindel i nätet" utav Kassa-klassen, eftersom den är en sådan central del utav ett köp i en dagligvaruhandel-butik, men genom att flytta ut själva transaktionen till en egen klass, samt att lägga ut hela lagret på en egen klass som uppdateras när en transaktion har genomförts så flyttades en del bort ifrån kassa-klassen. Man kan dock säga att jag på gränsen till att ha två spindlar i nätet nu med både transaktion och kassa-klasserna, har ingen bra lösning på detta i nuläget då båda dessa är så centrala i sammanhanget.

Valen mellan att lägga en klass som ett attribut istället för en egen klass upplever jag ibland är svåra att avgöra. Till exempel skulle attributet hylla på klassen vara, istället vara en helt egen klass, det skulle underlätta underhållet om man t.ex. vill byta namn på alla hyllor i en sektion. Nu ligger rabatt som ett attribut på klassen Vara, det kanske också borde vara en helt egen klass för att enkelt kunna överskåda pågående rabatter, och enkelt kunna matcha ihop dom med en kundgrupp, men just nu fanns det ingen sådan kravspecifikation, men det är ett enkelt sätt att göra systemet mera enkelt att uppgradera i framtiden.

Systemsekvens diagrammet var den del jag hade svårast för och är inte helt säker på fall jag har uppfattat exakt vad det ska symbolisera. Min uppfattning är att det ska visa hur interaktionen mellan personer och system, och interaktionen mellan systemen, men jag upplever att mitt systemsekvens diagram vart väldigt litet, men ser inte riktigt hur jag ska kunna utöka det.