**Newtons rapportmall**

**”Titeln på arbetet”**

**Kurs:**

Fastighetsföretagande, grund

**Klass:** Fastighetsförvaltare 5

**Termin och år:** HT 2011

**Författare:** Peter Andersson

**Lärare:** Anders Carlson

Sammanfattning

Innehållsförteckning

[Sammanfattning 2](#_Toc54938749)

[Inledning 4](#_Toc54938750)

[Tillvägagångssätt 4](#_Toc54938751)

[Slutsatser och avslutande diskussion 5](#_Toc54938752)

[Referenslista 5](#_Toc54938753)

[Bilagor 5](#_Toc54938754)

Inledning

Att arbeta med systemutveckling i all dess form ställer inte enbart krav på att ha god kännedom i de tekniska aspekterna. Arbetet sker nästan uteslutande i koalition med både arbetskamrater, chefer och externa aktörer. Många arbeten sker numera även i projektformat. En systemutvecklare behöver därför fler egenskaper än att kunna hantera sin mjukvara. Det finns stort vetenskapligt stöd just för i vilken form arbete i grupp bäst bedrivs på och det finns viss best practice i vilket arbetssätt som passar vilken typ av arbete. Ett agilt arbetssätt lämpar sig till mindre projekt och ett av dessa är scrum (Sommerville 2016). Syftet med detta arbete är att utföra given uppgift utefter principer i ett agilt arbete, primärt kring hur scrum appliceras.

Tillvägagångssätt

För att utföra ett så bra arbete som möjligt har gruppen initialt diskuterat hur upplägget ska se ut vad gäller vilka spelregler vi har, vilka aktiviteter vi har samt tidsplaneringen för detta.   
  
  
**Spelregler**Vad gäller spelregler är det bestämt att vi sitter från 9-16 varje dag i samtal. 16:30 görs dagligen en sammanfattning på dagen och vad som ska göras inför nästkommande dag.  
Sedan tidigare har gruppen skapat ett set med spelregler som även är tänkta att efterföljas i detta arbete. Följande punkter togs då upp:  
  
**Kommunikation**Om en person i gruppen får svårigheter med sina uppgifter är det bra att kommunicera det till gruppen i tid.

**Hålla tider/anmäla frånvaro**Ha tålamod och förståelse för formatet vi jobbar i.Ha förståelse för att kommunikation inte är densamma vid ett fysiskt möte

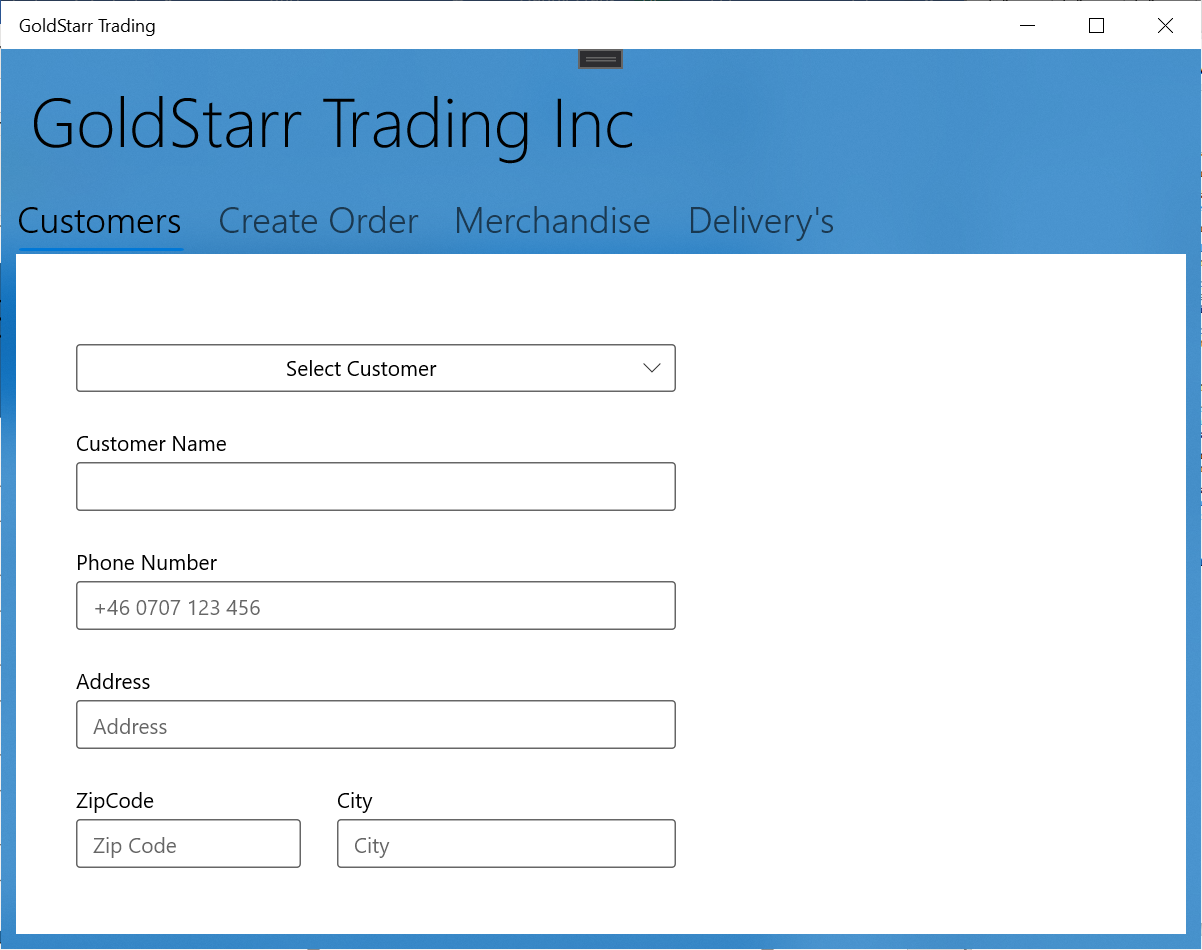
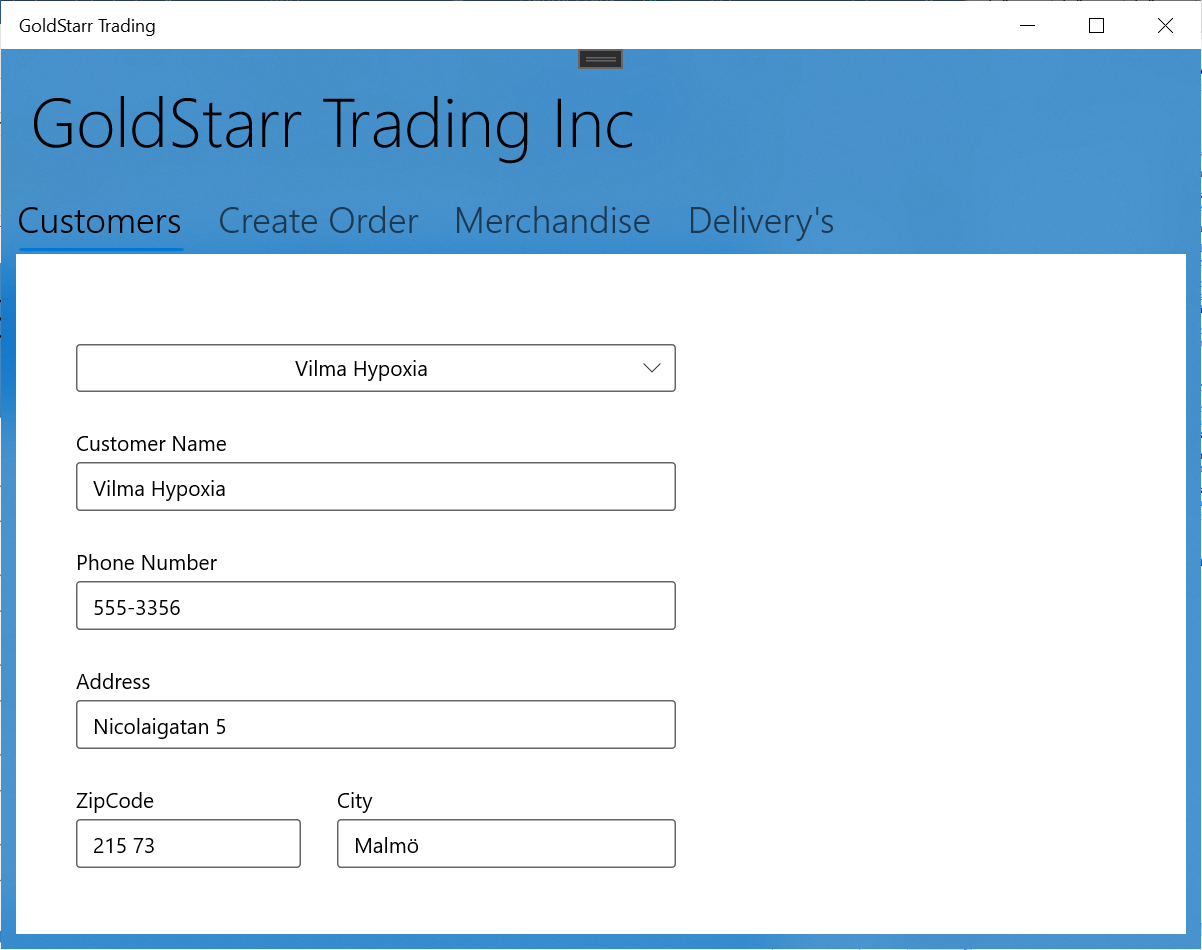
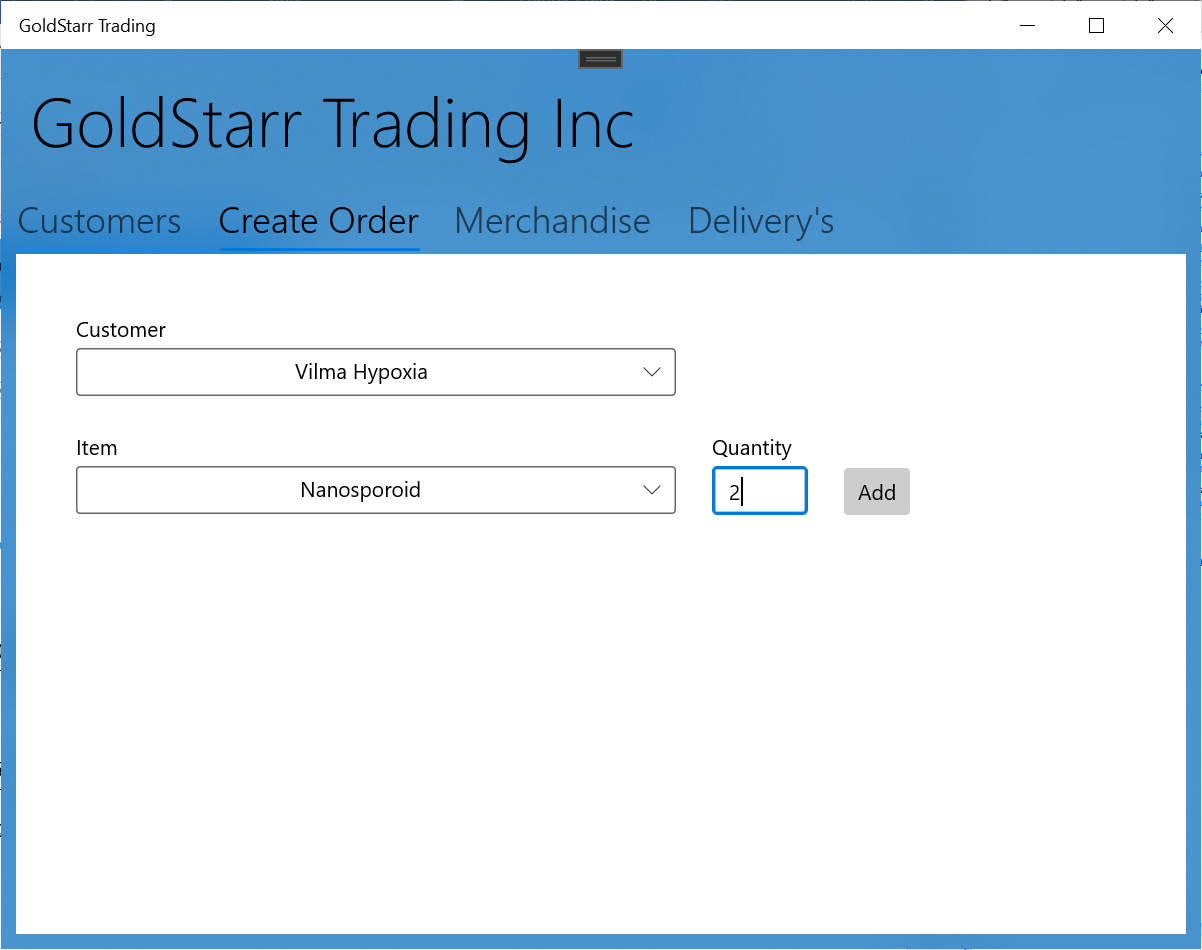
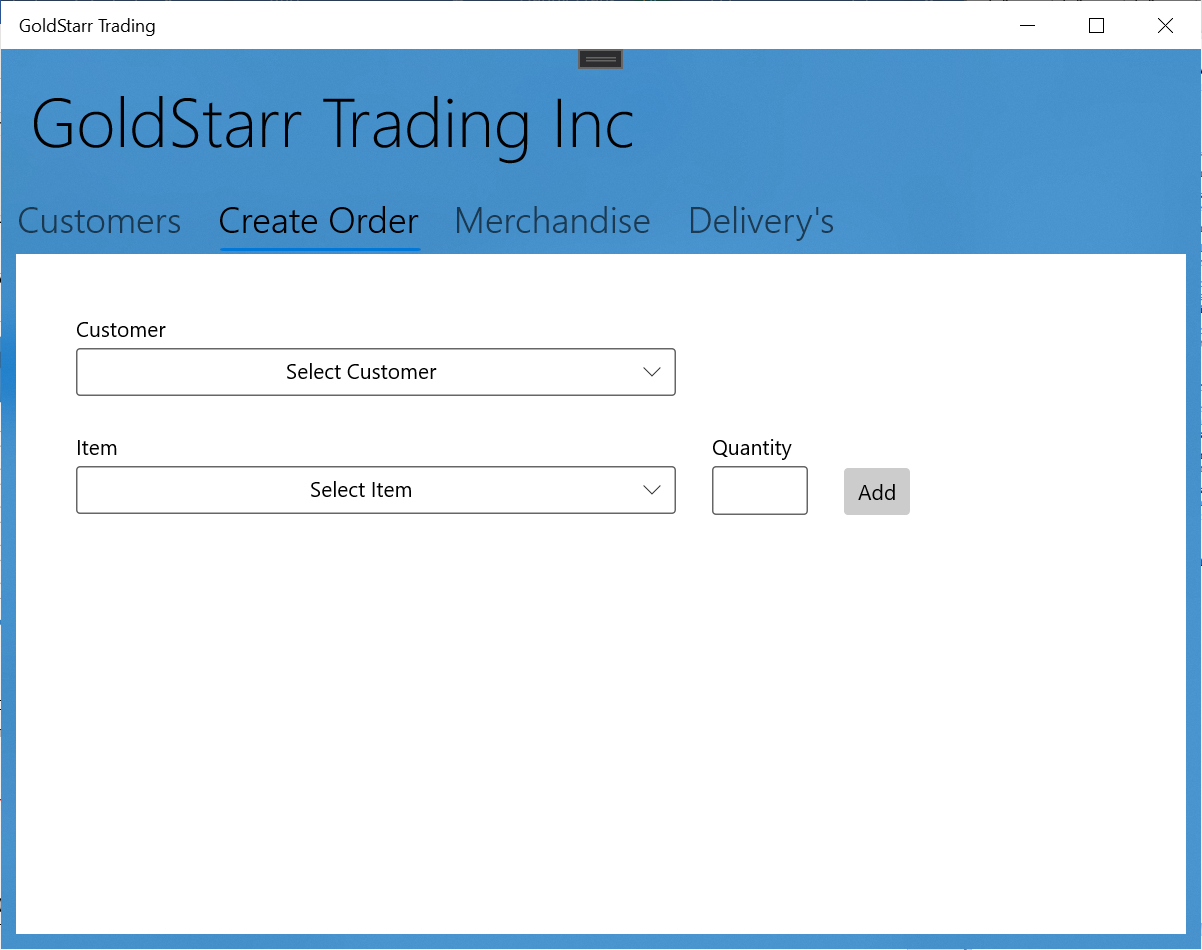
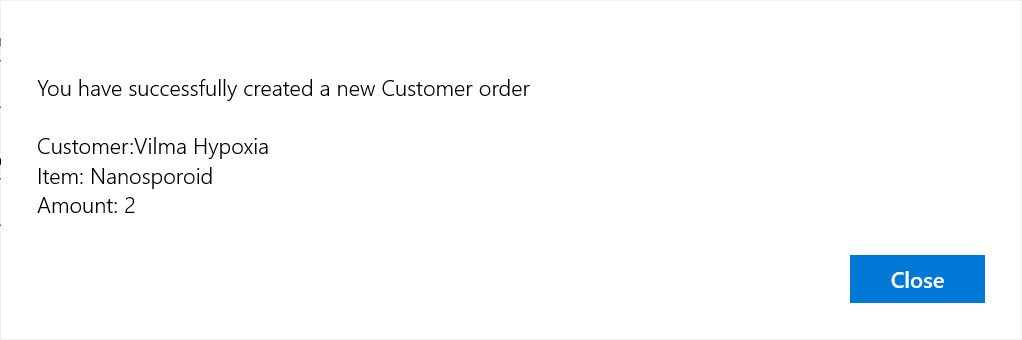
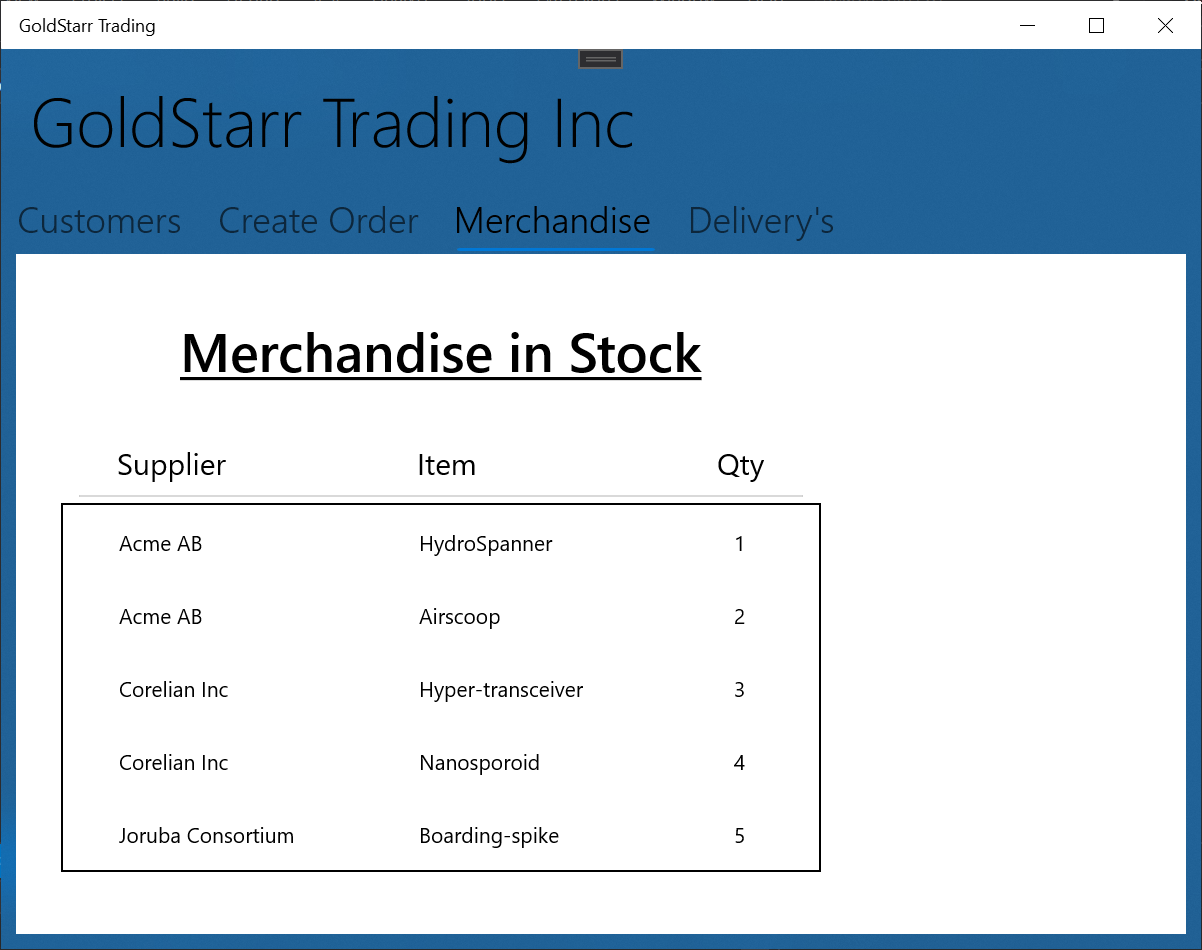
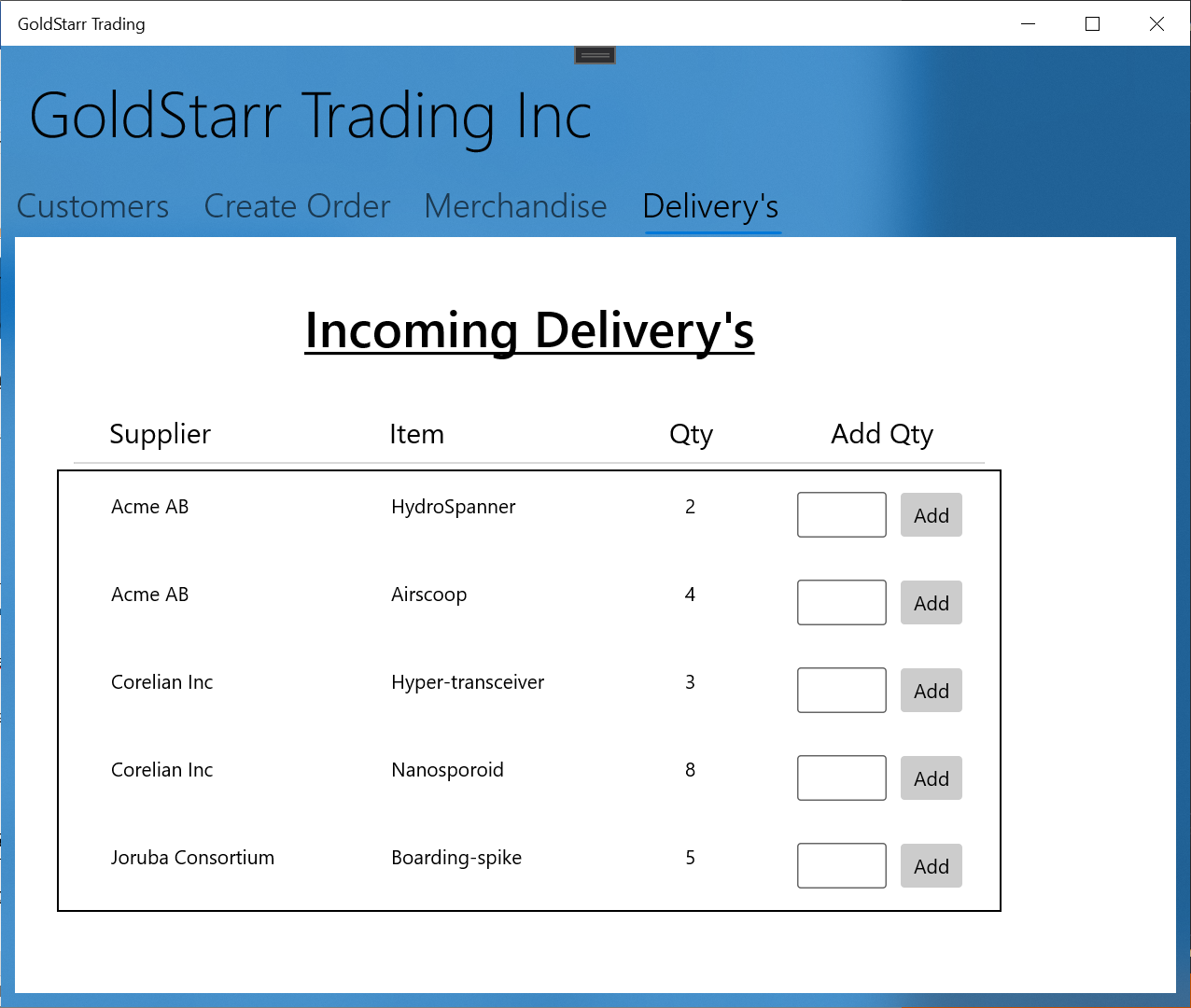
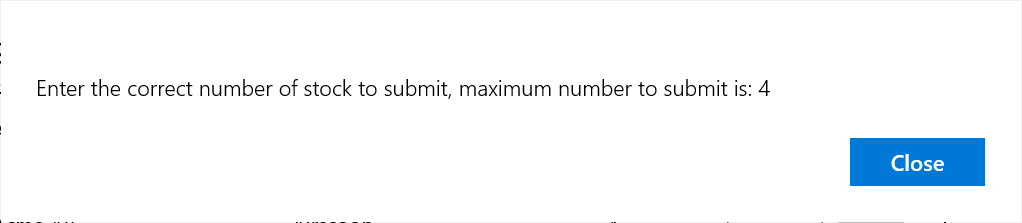
**Struktur**Att arbeta strukturerat och veta hur vi gör och vad vi gör, annars får vi stämma av med varandra

**Samarbeta**Dela information med varandra, förutsatt att detta sker inom rimliga gränser.

Om vi märker att dessa inte går att upprätthålla så är det viktigt att i tidigt skede kommunicera detta och agera enligt Newtons riktlinjer för projektarbete.

**Roll och ansvarsfördelning**Vad gäller roller och ansvarsfördelningså har gruppen valt att arbeta utefter ett agilt arbetssätt. Delar ur scrum-metodiken har applicerats i arbetet. Vi har dagliga möten vid samma tid där vi går igenom vad som har gjorts och vad som behöver göras. Arbetet är inte strikt uppdelat efter rollen utan alla i gruppen gör det som behöver göras.  **Nedbrytning av projektet**För att få en överskådlig bild av omfattningen av projektet är det viktigt att dela upp projektet i mindre delar. Vi har valt att dela upp projektet dels i de 2 olika releaserna som kommer att göras. Vi delar även in arbetet i dels en del som rör rapportskrivning och planering **WBS**Utifrån den diskussion som gruppen har haft har följande WBS skapats. Vi har valt att dela upp denna utefter de två olika releaserna. Vad gäller WBS:en för andra releasen så är inte denna nedbryten i lika många delmoment som release 1 då vi i planeringsfasen inte vet vilken funktionalitet release 2 ska ha. **Tidsplanering**Projektet pågår under 3 veckors tid. Efter den WBS som har gjorts har ett gantt-schema skapats över de aktiviteter som skall göras och den estimerade tiden för dessa.  
  
  
**Utförande  
  
Release 1**Uppdragsgivaren (project owner) önskar en applikation för att kunna hålla koll på sitt butikslager för en nystartad butik. Det finns en liten kundbas och leverantörer. På längre sikt finns det ett intresse av att expandera denna butik, bland annat med att även utveckla en onlinebutik. Initialt ställs följande krav på applikationen:

* En lista över nuvarande kunder
* En vy för att lägga till en ny order för en kund
* En lista över alla nuvarande produkter
* En vy som listar alla produkter och hur mycket som finns på lager
* En vy för att kunna hantera en leverans av produkter till lagret

Då företaget har planer på att expandera dels vad som finns för önskad funktionalitet inför release 2 men även framtiden, finns en del element att ta hänsyn till. Den applikation som utvecklas bör rimligtvis kunna skalas upp allteftersom företaget expanderar och fler funktioner behövs för applikationen. Det är därför av stor vikt i planeringsfasen att skapa ett användargränssnitt där det tagits hänsyn till att fler funktioner tillkommer. Det är även viktigt att klassindelningen och hur koden byggs upp underlättar för att kunna addera funktionalitet till applikationen.  
Nedan syns användargränssnittet för applikationen  
  
   
  
Första fliken visar företagets kunder. Överst finns en kombobox där användaren kan se de kunder som finns. Genom att klicka på respektive kund kan användaren få fram uppgifter om kunden i fråga.  
  
Under fliken create order kan användaren genom två komboboxar välja en kund välja den produkt som kunden önskar köpa. Till höger om komboboxen för val av produkter kan användaren skriva in antal som kunden önskar köpa. När användaren trycker på add kommer ett pop up fönster dyka upp som berättar vilken kund ordern lagts på, vilken produkt och antal. Skulle det vara så att lagret inte är tillräckligt så kommer ett felmeddelande berätta att ordern inte går att genomföra.  
  
  
  
Under fliken merchandise visas lagret för de produkter som företaget säljer. När En order skapas töms lagret på motsvarande mängd som ordern är på   
  
  
Under fliken Delivery´s visas de produkter som kommit in som ska läggas till i merchandise. Under Qty visas det hur många produkter som kommit in och under add Qty adderar användaren dessa till sitt lager, vilket uppdaterar antalet i merchandise. De kan vara så att företaget fått fler produkter än vad de beställt eller av någon anledning inte vill ta in alla inkommande produkter i lagret. Anvädaren har då möjlighet att inte addera hela inkommande beställningen till lagret. Skulle dock användaren försöka lägga till fler produkter till lagret än vad som kommit in så kommer ett felmeddelande att visas som visas nedan.   
  
  
  
  
**Release 2**  
Inför release 1 diskuterades vikten av skalbarhet i applikationen då det finns önskemål om att addera ny funktionalitet allteftersom företaget expanderar. Av de krav som project owner har arbetsgruppen diskuterat fram vad som är rimligt att hinna med att implementera inom det scope som finns vad gäller tid och funktion. Det arbetsgruppen har kommit fram till diskuteras vidare med project owner.  
  
  
  
Slutsatser och avslutande diskussion

Referenslista

Sommerville, I. (2016). *Software Engineering* (10:th Edition), Addison-Wesley

Bilagor