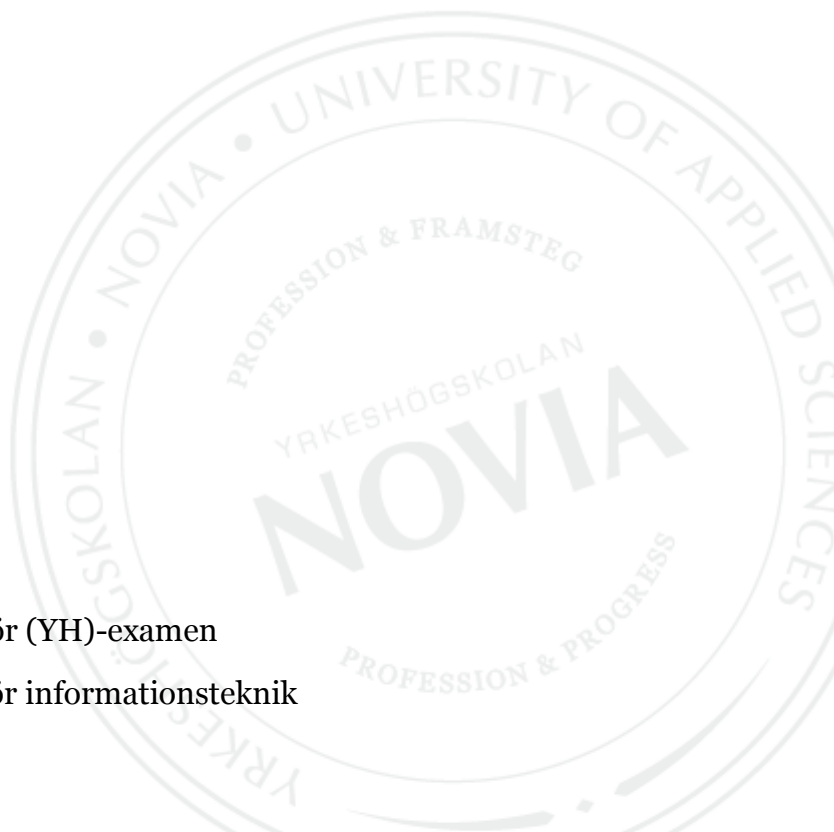




# **Modul för att skapa och redigera offertutskrifter**

Kim Nystrand

Examensarbete för ingenjör (YH)-examen  
Utbildningsprogrammet för informationsteknik  
Vasa 2012





## BACHELOR'S THESIS

Author: Kim Nystrand  
Degree programme: Information Technology  
Supervisors: Susanne Österholm

Title: *Module to create and edit quotation prints*

---

Date 28.10.2012

Number of pages 22

---

### Summary

This thesis work was made for the IT-company Ecraft in Vasa. The work was a part of a quote calculation program that KWH Pipe had ordered from Ecraft. The goal was to create a module that created a quote print, which could be edited before saving. The module is a part of a Sharepoint application with Web Parts. The module is made using Silverlight controls and the programming language is C#. For searches in the database LINQ was used and to send data between the server and the Silverlight module WCF was used. The application is currently under testing and is therefore not a finished product.

---

Language: Swedish

Key words: quote, Silverlight, C#

---

## OPINNÄYTETYÖ

Tekijä: Kim Nystrand  
Koulutusohjelma ja paikkakunta: Tietotekniikka, Vaasa  
Ohjaaja: Susanne Österholm

Nimike: *Moduuli tarjoustulosteiden luomiseksi ja muokkaamiseksi.*

---

Päivämäärä 28.10.2012

Sivumäärä 22

---

### Tiivistelmä

Tämä opinnäytetyö on tehty IT-yhtiö Ecraft:lle Vaasassa. Työ oli osa tarjouslaskentaohjelmasta joka KWH Pipe oli tilannut Ecraft:lta. Tavoitteena oli luoda moduuli joka loi tarjoustulosteen, joka olisi muokattavissa ennen tallennusta. Moduuli on osa Sharepoint-sovellusta jossa on käytetty Web Parts:ia. Moduulissa on käytetty Silverlight-kontrolleja ja ohjelmointikieli on C#. Tietokantahakuihin on käytetty LINQ:ta ja datan lähettämiseksi Silverlight-moduulin ja palvelimen välillä WCF on käytetty. Sovellusohjelma on parhaillaan testausvaiheessa eikä siis vielä ole valmis tuote.

---

Kieli: ruoti      Avainsanat: Tarjous, Silverlight, C#

---

# Innehållsförteckning

Sammanfattning

Summary

Tiivistelmä

Innehållsförteckning

Ordförklaringar

1 Inledning .....	1
1.1 Uppdragsgivare .....	1
1.2 Bakgrund .....	2
1.3 Uppgift .....	5
2 Språk och tekniker.....	6
2.1 .NET .....	6
2.1.1 C# .....	6
2.1.2 XAML .....	7
2.1.3 LINQ .....	7
2.1.3 Devepress .....	8
2.1.4 WCF .....	8
2.2 Silverlight .....	9
2.2 Visual Studio 2010 .....	10
3 Utförande.....	11
3.1 Planering .....	11
3.2 Grafiska gränssnittet.....	12
3.2.1 Popup-fönstret .....	12
3.2.2 Silverlight-sidan .....	14
3.3 Textredigeringen .....	15
4 Resultat.....	20
5 Diskussion .....	20
6 Källförteckning .....	22

## Ordförklaringar

Web Parts:	Web Parts är en ASP.NET server-kontroll som låter användare modifiera innehållet, utseendet och beteendet av webbsidor direkt från webbläsaren.
Sharepoint:	Sharepoint är en plattform för utveckling av webbapplikationer. Den låter användare skapa webbplatser och hantera dokument.
Kodfaktorisering	Är en teknik för att tydliggöra kod, minimera kod som finns dubbelt på flera ställen, bryta ner stora klasser till mindre osv.
MemoryStream	MemoryStream är en klass i namnrymden System.IO. Innehåller data sparad som en "unsigned" byte-räcka som finns i minnet.

# 1 Inledning

## 1.1 Uppdragsgivare

Ecraft är ett 1999 grundat IT-expertföretag som idag har ca 85 anställda på tre orter – Esbo och Vasa i Finland, samt Malmö i Sverige. Omsättningen ligger på ca 10 miljoner euro.

Ecraft är en Microsoft Gold Certifierad partner, vilket betyder att Ecraft måste ha ett visst antal anställda som är certifierade i att använda Microsofts produkter. Detta medför att Ecraft får billigare licenser vid köp av Microsofts produkter samt att de får teknisk support av Microsoft 24 timmar i dygnet. Detta medför att Ecraft kan ge bättre hjälp åt sina egna kunder och ha en bättre kundrelation. 2008 valdes Ecraft till årets Microsoft partner i Finland.

Företaget är partner med Lawson Service och Lawson Product. Lawson är ett företag som levererar affärssystemlösningar. I ett affärssystem finns ofta moduler för order- och lagerhantering, produktionsplanering, projektplanering, inköp m.m. Man kan säga att ett affärssystem är ett system där man kan samla all data, i stället för att ha det utspritt i flera program. Genom att vara produkt- och servicepartner kan Ecraft erbjuda och underhålla Lawsons produkter åt sina egna kunder.

Företaget är partner med QlikView, vilket betyder att de är återförsäljare av QlikView. QlikView är ett beslutsverktyg, där beslutsverktyg är ett samlingsbegrepp för färdigheter, teknologier, applikationer, processer och metoder för organisationer att bättre förstå sin verksamhet. Med QlikView kan man t.ex. visualisera data med tekniskt avancerad grafik.

Ecraft är också HP Gold Preferred partner, vilket betyder att Ecraft fått utbildning i att förse organisationer med kompletta och kundanpassade lösningar från HP.

Ecraft är ett mjukvaruföretag som erbjuder skräddarsydda lösningar, integrationer som för samman olika datorsystem och program fysiskt eller funktionellt. Ecraft erbjuder också konsultering åt företag i hur man bäst använder IT för att möta deras affärsmål.

Ecrafts slogan är: ”We create less software”, som kan ses i figur 1. Ecraft anser att det idag finns för många svåra, komplicerade och frustrerande informationssystem. Därför vill de att den enskilda användaren behöver så lite mjukvara som möjligt, så att användaren kan göra sitt jobb utan en massa problem med mjukvaran. (Företagspresentation, u.å)



Figur 1. Ecrafts logo samt slogan.

## 1.2 Bakgrund

Ecraft hade gjort ett offertberäkningsprogram åt KWH Pipe. Programmet har fem olika delar som är kund, projekt, produkt, installation och summering. Programmet är gjort i Sharepoint med Web Parts. I alla fem delarna fyller man i information som sparas i en databas.

Search Projects    Create Project    Configuration    Search this site...				
Drag a column header here to group by that column				
Customer	Name	ToL	Country	Status
Billerud			Tokelau	Calculation
Boliden	test			Calculation
Boliden	123456	Pneumatic	Afghanistan	Calculation
Boliden	Testprojekt	Chemical pipe	Afghanistan	Calculation
Boliden	Aitik 36	Wet slurry	Sweden	Calculation
Cembrit Oy	Cembrit	Pneumatic	Finland	Calculation

Figur 2. Vyn över skapade projekt.

På första sidan listas alla projekt (dvs. offerter) som blivit skapade. Vill man göra en ny offerberäkning klickar man på ”Create Project”, som visas i översta menyn i figur 2. I kunddelen kan man skapa/välja kund. Väljer man att skapa en ny kund kommer man till en vy där man fyller i kundinformation som kan ses i figur 3.



Figur 3. Insättning eller sökning av kund.

Om kunden redan finns i databasen kommer man in i ett icke-redigerbart läge där informationen visas.

I projektdelen anger man information om projektet, bl.a. namn, plats, land, kontaktpersoner osv. Projektdelen kan ses i figur 4.

Figur 4. Insättning och editering av projektinformation.

I Figur 5 visas produktdelen där man väljer produkter (dvs. rör), samt anger deras längd. Priset på produkterna visas direkt en produkt blir vald. Det går också att klicka på länken "Show layers...", som visar vilka lager produkten innehåller, deras densitet, tjocklek m.m.

Search Projects    Create Project    Configuration

KWH BYG

Customer    Project    **Product**    Installation    Summary

Double click a row to open the edit view.

Name	Length (m)	Material cost (€/m)	Material cost (€)
WehoAntistatic PN10 250/215	6,50	42,91	278,91
Product: <input type="text" value="WehoAntistatic PN10 250/215"/> Changed by: Nystrand, Kim (external) Length (m): <input type="text" value="6,50"/> Changed date: 13.10.2012 7:38:57 Metric weight (kg/m): 12,38    Created by: Nystrand, Kim (external) Weight (kg): 80,49    Created date: 13.10.2012 7:38:57 Material cost (€): 278,91 Show layers... <input type="button" value="Save"/> <input type="button" value="Cancel"/> <input type="button" value="Delete"/>			
WehoAntistatic PN10 90/77	5,00	6,01	30,05
	11,50	48,92	308,95

Add product

   [Go back](#) or

Figur 5. Val av produkter.

I följande del finns det möjlighet att ange olika tillägg till offerten. Med detta menas t.ex. kopplingsutrustning, eventuella montörer etc. samt kostnader för dessa. I delen som går att se i figur 6, kan man välja vilka olika kopplingar som behövs samt deras typer och antal. Användaren matar själv in priset per produkt.

KWH BYG

Customer    Project    Product    **Installation**    Summary

Double click a row to open the edit view.

Name	Length (m)	Material Cost (€)	Other Cost (€)																						
WehoAntistatic PN10 250/215	6,50	326,91	450,00																						
<table border="1"> <thead> <tr> <th>Joint</th> <th>Type</th> <th>Quantity</th> <th>Price</th> <th>Cost</th> <th></th> </tr> </thead> <tbody> <tr> <td><input type="text" value="Mechanical Couplings"/></td> <td><input type="text"/></td> <td><input type="text" value="6"/></td> <td><input type="text" value="8,00"/> €/pcs</td> <td>48 €</td> <td>Delete</td> </tr> </tbody> </table> Add new jointing <table border="1"> <thead> <tr> <th>Description</th> <th>Quantity</th> <th>Unit cost</th> <th>Cost</th> <th></th> </tr> </thead> <tbody> <tr> <td><input type="text" value="KWH Engineer support"/></td> <td><input type="text" value="1"/></td> <td><input type="text" value="450,00"/> €</td> <td>450 €</td> <td>Delete</td> </tr> </tbody> </table> Add other cost <input type="button" value="Save"/> <input type="button" value="Cancel"/>				Joint	Type	Quantity	Price	Cost		<input type="text" value="Mechanical Couplings"/>	<input type="text"/>	<input type="text" value="6"/>	<input type="text" value="8,00"/> €/pcs	48 €	Delete	Description	Quantity	Unit cost	Cost		<input type="text" value="KWH Engineer support"/>	<input type="text" value="1"/>	<input type="text" value="450,00"/> €	450 €	Delete
Joint	Type	Quantity	Price	Cost																					
<input type="text" value="Mechanical Couplings"/>	<input type="text"/>	<input type="text" value="6"/>	<input type="text" value="8,00"/> €/pcs	48 €	Delete																				
Description	Quantity	Unit cost	Cost																						
<input type="text" value="KWH Engineer support"/>	<input type="text" value="1"/>	<input type="text" value="450,00"/> €	450 €	Delete																					
WehoAntistatic PN10 90/77	5,00	30,05	0,00																						
	11,50	356,95	450,00																						

[Go back](#) or

Figur 6. Val av installationer.

Slutligen finns det en summeringsdel där alla produkter och deras kostnader listas, samt en vy där man för varje produkt kan se vilka extra specifikationer och detaljer som angetts för offerten, se figur 7. Genom att klicka på ”Margins and selections” (se figur 8) finns

möjligheten att avmarkera produkter och installationer som slutligen inte behövs, och att ange t.ex. rabatter på produkterna och dylikt.

Double click a row to see row details.

Name	Length (m)	Material Cost (€)	Other Cost (€)	Conversion and transfer (€)	Margin (€)	Sales Price
WehoAntistatic PN10 250/215	6,50	326,91	450,00	0,00	0,00	326,91
WehoAntistatic PN10 90/77	5,00	30,05	0,00	0,00	0,00	30,05
	11,50	356,95	450,00	0,00	0,00	356,95

Project status:  Margins and selections  [Go back](#)

Figur 7. Summeringen av valda produkter.

Product	Cost (€)	Total Conversion (€/kg)	Total Conversion (%)	Transfer Fee (%)	Sales Margin (%)	Total Price
<input checked="" type="checkbox"/> WehoAntistatic PN10 250/215	278,91	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	278,91 €
<input checked="" type="checkbox"/> Mechanical Couplings	48	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	48 €
<input checked="" type="checkbox"/> KWH Engineer support	450	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	450 €
<input checked="" type="checkbox"/> WehoAntistatic PN10 90/77	30,05	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	30,05 €
	806,95					806,95 €

Figur 8. Redigering av produkter och installationer.

### 1.3 Uppgift

Uppgiften var att det i summeringsdelen i programmet skulle gå att skapa en offertutskrift för en offert. Offertutskriften skulle dels innehålla informationen som man matat in på de föregående delarna, dels allmänna regler för köp av produkter och tjänster av KWH Pipe.

Offertutskriften skulle också kunna redigeras, ifall t.ex. moms eller faktureringsstid ändras. KWH Pipe gav offertmallar på svenska, finska och engelska, så på något sätt skulle funktionen att välja språk när man skapade utskriften finnas. När offertutskriften var klar skulle den skickas per epost till användaren som skapat offerten. Offertutskriften skulle också sparas till databasen, så att man i ett senare skede kan gå in och ändra i offertutskriften och återsända den.

## 2 Språk och tekniker

### 2.1 .NET

.NET-ramverket är en plattform för att bygga system och program främst för Microsoft Windows, men också för andra operativsystem som Mac OS X och Linux. Plattformen stöder programmeringsspråk som t.ex. C#, Visual Basic och F#.

Man kan säga att .NET är en samling basklasser som används av alla språk som stöds av ramverket. Med hjälp av basklasserna kan man använda ASP.NET för att skapa webbsidor, WPF för att bygga grafiska användargränssnitt osv. .NET sköter också om exekveringen av program. (Troelsen, 2012, s. 3–5)

#### 2.1.1 C#

C# är ett av de programmeringsspråk som framtagits speciellt för att bygga applikationer i .NET. C# har tagit de bästa funktionerna från språk som C, C++, Java och Visual Basic för att få ett så bra språk som möjligt. Syntaxen i C# liknar väldigt mycket Java.

Eftersom syntaxen i C# är väldigt enkel, har C-, C++- och Javaprogrammerare inga problem med att skriva C#-kod. Språket är också objektorienterat. Det stöder inkapsling, arv och polymorfism, vilket medför att det blir lättare att ändra och omstrukturera en applikation. (Bakharia, 2002, s.3)

Med C# kan man skapa grafiska användargränssnitt med Windows Forms och WPF, dvs. program som visar knappar, fönster, menyer m.m. vilka man har för att användaren skall kommunicera med programmet. För detta finns inbyggda verktyg i C#-språket. Språket stöder också ”multithreading”, vilket betyder att man kan skapa program som kan ha flera aktiviteter pågående samtidigt, t.ex. visa en rörlig bild samtidigt som en användare kan mata in data i ett fönster. (Skansholm, 2008, s. 6)


## 2.1.2 XAML

XAML är en förkortning av ”Extensible Application Markup Language” och det är baserat på XML. XAML är skapat av Microsoft och används i både Silverlight och WPF för att skapa grafiska användargränssnitt. XAML är ett skiftlägeskänsligt språk.

XAML används i Silverlight för att definiera bl.a. det grafiska användargränssnittet för applikationer. Varje XAML-fil har oftast en code-behind-fil där all programlogik finns. I code-behind-filerna använder man .NET språk som C# och VB.NET. I kodexempel 1 visas hur en knapp i ett grafiskt användargränssnitt definieras med XAML-kod. (Snow, 2009, s. 19f)

*Kodexempel 1. XAML kod för hur en knapp definieras.*

```
<Grid>
  <Button Content="Button" Width="70" Height="30" />
</Grid>
```



## 2.1.3 LINQ

LINQ står för ”Language Integrated Query”. Det är en del av .NET-ramverket, och är en samling bibliotek för att arbeta med data. Med data menas XML-dokument, CSV-filer, databaser, olika samlingar m.m. Det som gör LINQ så populärt är att det låter utvecklare söka genom data genom att använda C#-språkets syntax oberoende av typen av datakälla. Om man inte använder LINQ blir man tvungen att använda olika tekniker för att söka genom olika typer av datakällor. (Magennis, 2010, s. 1f)

Kodexempel 2 visar hur en sökning i databasen görs enligt producentens id och sparas till en lista. Först anges var källan för datan är, sedan hur sökningen filtreras, i detta val enligt ”producerId”, och till sist väljer man datan genom ”select”.

*Kodexempel 2. En simpel LINQ-select sats.*

```
public static List<Contract> GetContracts(int producerId)
{
    var query = from c in DataContext.Contracts
                where c.ProducerId == producerId
                select c;

    return query.ToList();
}
```

Största målet med LINQ är att det skall bli enklare för programmerare att arbeta med relationsdatabaser. (Klein, 2008, s. 3)

### 2.1.3 Devexpress

Devexpress är ett företag som funnits sedan 1998 och erbjuder XAML-kontroller och verktyg för WinForms, ASP.NET, WPF och Silverlight. Förutom att Devexpress-verktygen innehåller nästan alla kontroller som redan finns i ASP.NET, Silverlight osv., erbjuder de också kontroller för rapporter, textredigering, rättstavning, kartor etc.

Kontrollerna är speciellt byggda för att förbättra produktiviteten för användarna och att leverera övertygande, högpresterande och funktionella affärslösningar. Det som är bra med dessa kontroller är att de är byggda så att programmeraren behöver programmera mindre. De sköter det mesta automatiskt genom kontrollens egenskaper och har väldigt många smarta, inbyggda funktioner. (About Devexpress, u.å)

### 2.1.4 WCF

WCF är en förkortning av ”Windows Communication Foundation”. Det är en teknik från Microsoft för att bygga tjänster (eng. Services) och det är byggt på .NET-ramverket, vilket förenklar utvecklingen av system. Man kan säga att WCF tillåter applikationer i en distribuerad miljö att kommunicera med varandra. Det tillåter också utvecklare att bygga pålitliga och säkra lösningar som integreras över olika plattformar och samverkar med existerande program. (Liu, 2010, s. 14)

En WCF-tjänst är uppbyggd enligt en ABC-metod.

- A står för Address: Var är tjänsten?
- B står för Binding: Hur kommunicerar man med tjänsten?
- C står för Contract: Vad kan tjänsten göra?

När man skall bygga en WCF-tjänst så fungerar det också enligt ABC-metoden, alltså enligt en tre-steps-metod. Först definierar man kontraktet och implementerar det i tjänsten. Sedan väljer man eller definierar en tjänstebindning och väljer en transportmetod tillsammans med kvalitet, säkerhet och andra egenskaper. Till sist sätter man en slutpunkt för kontraktet genom att binda det till en nätverksadress. (Introduction to Building Windows Communication Foundation Services, u.å)

## 2.2 Silverlight

Silverlight är en webbaserad plattform för att bygga tillämpningsprogram som körs i webbläsaren. Det som gör att det är webbaserat är ett insticksprogram som körs i webbläsaren, och som installeras på den lokala hårddisken på samma sätt som man installerar Adobe Flash för att köra flash-baserade animationer på hemsidor. Insticket stöder vektorbaserad grafik, animationer, full videointegration och högupplöst video.

Silverlight-spelaren är en helt fristående insticksmodul, vilket betyder att den inte alls behöver .NET-ramverket varken på klienten eller på servern. XAML-kod används för att skapa användargränssnittet för Silverlight-applikationen. XAML-koden laddas ned till webbläsaren och körs i Silverlight-insticksmodulen. När XAML körs i Silverlight-spelaren i webbläsaren, får man tillgång till Silverlight-objekten via JavaScript-kod som finns på HTML-sidan. (Rader, Hinkson, Beres, 2008, s.1–2)

Positiva saker med Silverlight:

- Man kan köra Silverlight-applikationer på flera olika system och i flera olika webbläsare.

- Det stöder uppspelning av video- och ljudfiler på PC och MAC.
- Installationspaketet är mindre än 2 MB.
- Samma XAML-kod som skapats i Silverlight kan användas i en WPF-applikation utan att den behöver ändras.

Negativa saker är:

- Om användaren har många andra mediaprogram som kräver mycket av bandbredden kan det påverka Silverlight-användarupplevelsen.
- I användarens webbläsarinställningar, kan man stänga av skript och kontroller. Dessa inställningar kan begränsa eller förhindra körningen av Silverlight-applikationer. (Dayley, 2008, s. 4–5)

## 2.2 Visual Studio 2010

Visual Studio 2010 är en integrerad utvecklingsmiljö (IDE), innehållande en samling med verktyg som underlättar utvecklingen av program. Visual Studio 2010 är utvecklat av Microsoft. Med VS kan man skapa flera olika typer av applikationer och komponenter (Mayo, 2010, s.4–5)

VS används främst för att skapa konsolprogram, program med grafiska användargränssnitt (eng. GUI), webbsidor, webbapplikationer m.m.

Visual Studio innehåller en kodredigerare som stöder ”IntelliSense”, vilket betyder att när kod skrivs så föreslår programmet vad som kan skrivas. Den har också stöd för kodfaktorisering. Andra inbyggda verktyg är formulärdesigner för att bygga grafiska användargränssnitt, webbdesigner, klassdesigner och databasschemadesigner. Det går också att installera insticksprogram som förbättrar programmet.

Programmet stöder programmeringsspråk som: C/C++, VB.NET & C#. Även andra språk går att använda genom att installera ytterligare språkpaket. VS stöder också XML/XSLT, HTML, XHTML, Javascript och CSS. VS kommer i flera versioner: Express, Professional, Premium och Ultimate. (Microsoft Visual Studio, u.å.)



## 3 Utförande

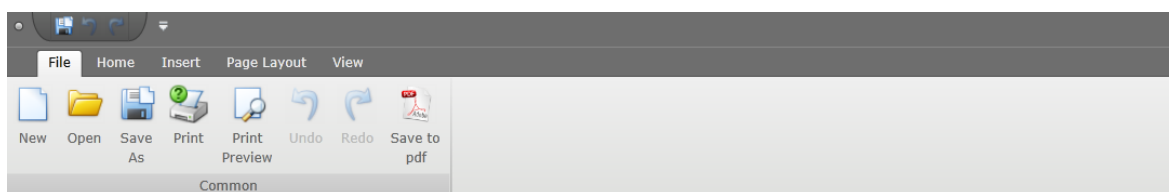
### 3.1 Planering

Eftersom Ecraft är ett partnerföretag till Microsoft, och den befintliga applikationen var gjord i Sharepoint med Web Parts, så skulle även offertutskriftsdelen göras med antingen ASP.NET, WPF eller Silverlight. Det gjorde att alternativen inte var så många som om det t.ex. skulle göras ett helt fristående program.

Det som planerades innan programmeringen startade var vilken kontroll som skulle användas för att det i programmet skulle gå att redigera en Microsoft Word-mall av en offert.

Efter lite sökande på Internet hittades en kontroll som Devexpress erbjuder, en Silverlight kontroll med namnet RichEdit. Devexpress erbjuder på sin hemsida demonstrationer av alla sina kontroller, så det går att se hur de ser ut och vilka funktioner som finns.

RichEdit var ett mycket bra val p.g.a. att det fanns möjligheten att kombinera textredigeraren med en s.k. ”BarManager” som också finns bland utbudet av kontroller från Devexpress. ”BarManagern” ger menyer och flikar med olika alternativ för redigering av text. Med dessa kombinerade skulle sidan få ett utseende av en nedbantad version av Microsoft Word. (Se Figur 9.)

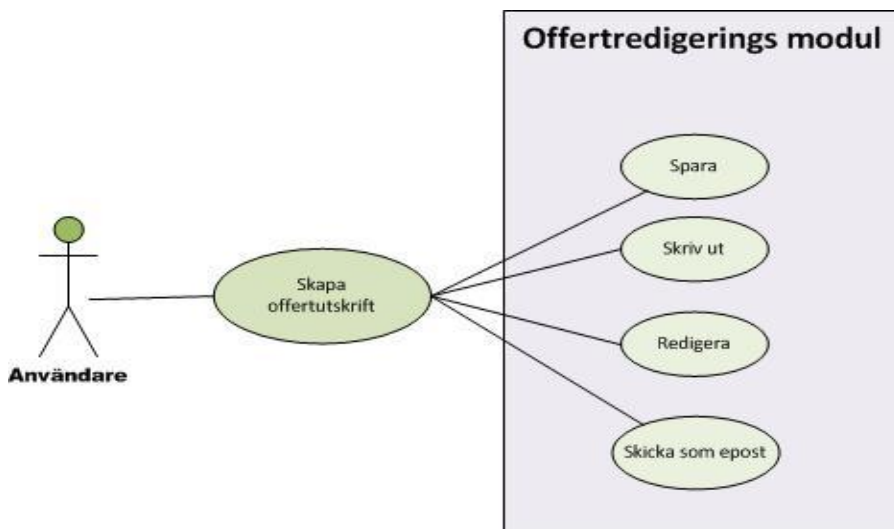


*Figur 9. Bild av menyn i Silverlight-sidan.*

Eftersom RichEdit var en Silverlight-kontroll skulle det skapas ett nytt Silverlight-projekt i det befintliga projektsamlingen, som är en Sharepoint-applikation med Web Parts. Man skulle också bestämma hur sidan skulle visas. Eftersom det tidigare visats liknande funktioner som popup-fönster, föll valet på att visa Silverlight-sidan i en ASP popup-kontroll. Där fanns två olika möjligheter. Första alternativet var att göra allt i summeringssidans kod, vilket skulle betyda att man bygger upp allt i popup-kontrollens

innehåll. Andra alternativet var att skapa en helt ny sida och ange var popup-kontrollens innehåll finns genom en adress och skicka med en projekt-id. Det senare valdes med motiveringen att det är lättare att ha all kod för Silverlight i ett eget projekt.

Desto mer planerades inte och de problem som dök upp under utvecklingen åtgärdades vartefter. I figur 10 finns ett Use-Case diagram över hur det är tänkt att textredigeraren ska fungera.



Figur 10. Use-case över hur redigeringen fungerar.

En användare kan skapa offertutskriften efter att alla produkter och installationer är valda. I Silverlight-textredigeringsmodulen finns då möjligheterna att spara offertutskriften, skriva ut den, redigera offertutskriften ifall någon rad skall tilläggas osv. eller skicka iväg offerten som epost.

## 3.2 Grafiska gränssnittet

### 3.2.1 Popup-fönstret

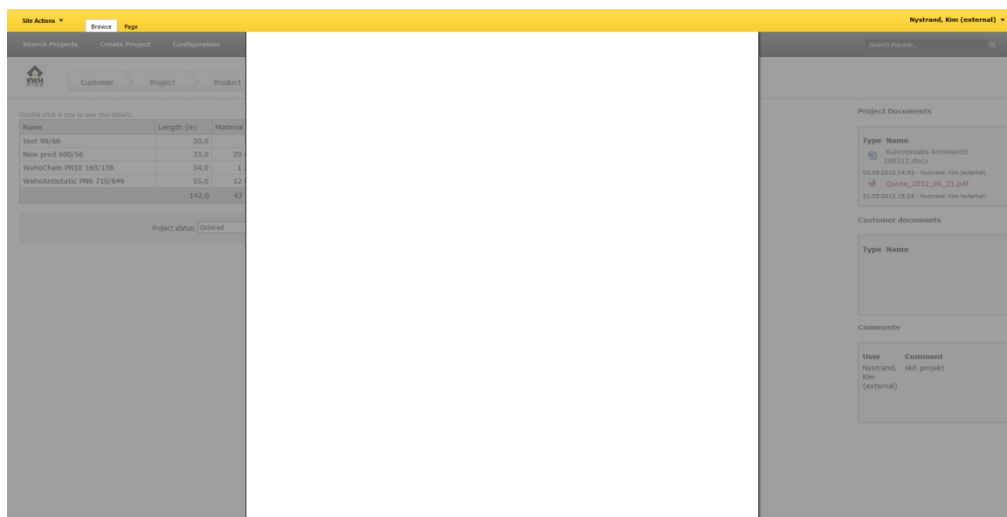
I summeringsdelen finns en knapp med namnet "Create quote" Denna knapp kallar på en ASPxPopupControl, som är en kontroll från Devexpress. Med popup menas att fönstret kommer upp ovanpå den befintliga sidan. De egenskaper som man anger i detta skede är hur popup-fönstret skall se ut. Höjd och bredd har angivits samt ifall det skall gå att ändra storlek på fönstret. Eftersom fönstret är en popup går det lätt att stänga genom att trycka på

knappen ”escape” eller klicka utanför fönstret. Inställningarna för popup-fönstret kan ses i kodexempel 3.

### Kodexempel 3. XAML-kod för popup-fönstret.

```
<dx:ASPxPopupControl ID="ASPxPopupControl1" ClientInstanceName="ASPxPopupControl1" HeaderText="Offert"
    AllowDragging="true" AllowResize="true" Modal="true" runat="server" PopupHorizontalAlign="WindowCenter"
    PopupVerticalAlign="WindowCenter" Height="1000" Width="1000">
    <ContentCollection>
        <dx:PopupControlContentControl ID="PopupControlContentControl2" runat="server">

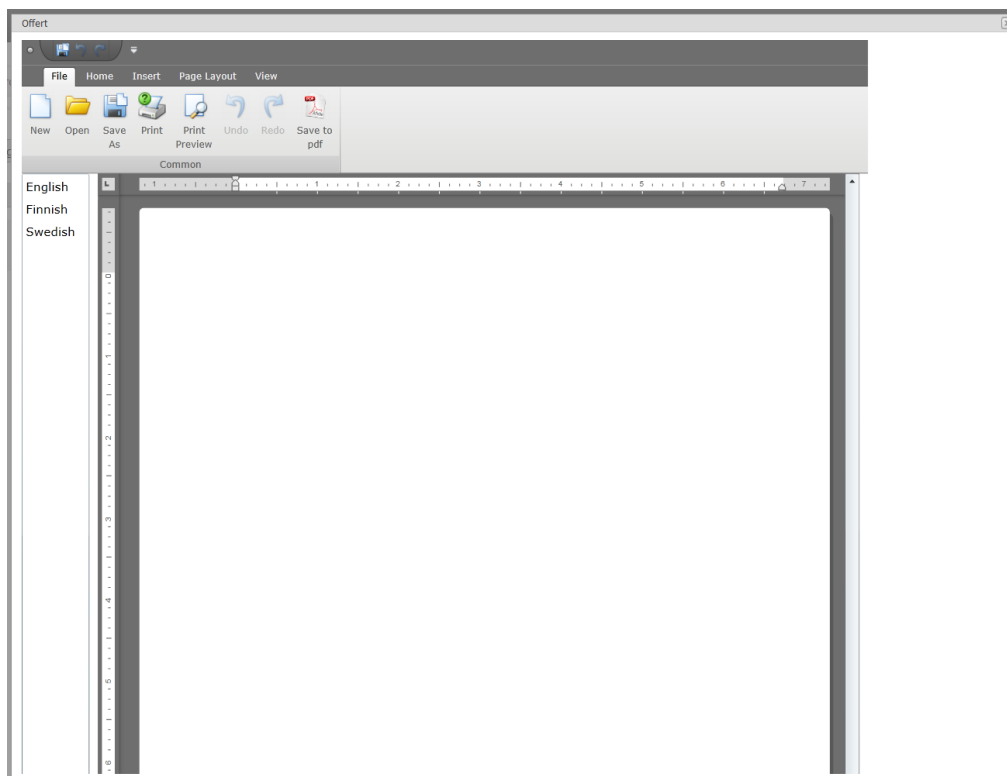
            </dx:PopupControlContentControl>
        </ContentCollection>
    </dx:ASPxPopupControl>
```



Figur 11. Screenshot av popup-fönstret utan innehåll.

För att något skall visas i fönstret behöver innehållsadressen (eng. Content URL) anges. Det görs i koden när summeringssidan laddas. Man skickar även med projektets id i adressen för att senare kunna hämta ut rätt information till offerten från databasen. Popup-fönstret utan innehåll visas i figur 11.

### 3.2.2 Silverlight-sidan



Figur 12. Screenshot av popup-fönstret med innehåll.

Sidan innehåller en menykontroll med namnet BarManager, som innehåller alla menyer och knappar högst upp på sidan (Figur 12). Det goda med menykontrollen är att man inte behöver implementera varje knapp och funktion, det sköts automatiskt genom referenserna man angivit.

Längst ner på sidan finns en RichEdit-kontroll, som visar offerutskriften. Den knyts till menykontrollen så att det går att använda alla funktioner på texten som skrivs i textfältet. För att kunna välja mellan olika språk av offerten skapades en ListBox, som finns till vänster om textfältet. Denna innehåller i dagsläget tre språk: svenska, finska och engelska. När sidan öppnas är textredigeringskontrollen tom, och vid val av språk laddas en offertmall in till textredigeraren.

I menyn är det bara knappen ”Save to PDF” som inte fanns från förut. Den skapades genom att i XAML-koden sätta in en ny knapp samt ange en ikon och en klickmetod. Knappen sparar dokumentet som en PDF-fil på fördefinierad plats på hårddisken, samt

skickar iväg offerten och allmänna villkor för offerten som bilagor till epost. Mottagaren är den inloggade användaren som skapat offerten.

### 3.3 Textredigeringen

Eftersom Silverlight-sidan körs på klienten var WCF ett måste för att kunna hämta data från databaser, serverns lokala hårddisk osv. Först skapas ett kontrakt, som heter IFPSService. Ett kodexempel för hur ett kontrakt skapas visas i kodexempel 4. Det är detta kontrakt som klienten måste gå via för att kunna kommunicera med servern, och det är här alla metoder definieras som klienten har möjlighet att kalla på. Sedan går det att i FPSService, som finns på servern implementera metoderna och därifrån komma åt databaserna och få ut data man behöver.

*Kodexempel 4. Exempel ur klassen IFPSService där metoderna till servern definieras.*

```
[ServiceContract]
public interface IFPSService
{
    [OperationContract]
    byte[] GetTemplate(string language);

    [OperationContract]
    FPS2.FPSData.FPSEntities.FPSProject GetProjectData(Guid projectId);

    [OperationContract]
    FPS2.FPSData.FPSEntities.FPSCustomer GetCustomer(Guid projectId);

    [OperationContract]
    List<FPS2.FPSData.FPSEntities.FPSProduct> GetProductData(Guid projectId);
}
```

I kodexempel 5 visas hur man får tillgång till serverns klasser och i kodexempel 6 visas en metod innehållande en select-sats i LINQ där en kund plockas ut från databasen.

Kodexempel 5. Exempel ur klassen FPSService.

```
public FPSCustomer GetCustomer(Guid projectId)
{
    Guid customerId = ProjectDAOUtils.Instance.GetCustomerIdForProject(projectId);
    FPSCustomer customer = CustomerDAOUtils.Instance.GetCustomer(customerId);

    return customer;
}
```

Kodexempel 6. GetCustomer-metod i serverns "Data access object"-klass där man gör sökningar i databasen och plockar ut den data man behöver.

```
[DataObjectMethod(DataObjectMethodType.Select, false)]
public FPSCustomer GetCustomer(Guid Id)
{
    FPS2Entities dc = new FPS2Entities(ConfigurationManager.ConnectionStrings[ConnectionStringKey].ConnectionString);
    var customer = (from c in dc.Customers
                    where c.Id == Id
                    select new FPSCustomer
                    {
                        Id = c.Id,
                        Name = c.Name,
                        Address = c.Address,
                        City = c.City,
                        PostalCode = c.PostalCode,
                        Contact1 = c.Contact1,
                        Contact2 = c.Contact2,
                        ModifiedBy = c.ModifiedBy,
                        ModifiedOn = c.Modified,
                        CreatedBy = c.CreatedBy,
                        CreatedOn = c.Created,
                        Phone = "N/A",
                        CountryId = c.Country == null ? (Guid?)null : c.Country.Id,
                        Country = new FPSCountry
                        {
                            Id = c.Country.Id,
                            ISO2Code = c.Country.CountryCodeIso2,
                            ISO3Code = c.Country.CountryCodeIso3,
                            Name = c.Country.Name,
                            NumericalCode = c.Country.CountryNumber
                        }
                    }
                    ).FirstOrDefault<FPSCustomer>();

    return customer;
}
```

Hämtningen av offertmallen som var ett Microsoft Word-dokument, gjordes så att först specificerades det i Sharepointens web.config var (på serverns lokala hårddisk) mallarna fanns (se kodexempel 7). Sedan kunde man med hjälp av klassen ConfigurationManager få ut adressen till mallen och få mallen som en Array med bytes, som ses i kodexempel 8.

*Kodexempel 7. En Web.config-fil med mallarnas adresser på serverns lokala hårddisken.*

```
<appSettings>
  <add key="TemplateSwedish" value="C:\temp\Swedish.docx" />
  <add key="TemplateFinnish" value="C:\temp\Finnish.docx" />
  <add key="TemplateEnglish" value="C:\temp\English.docx" />
  <add key="siteUrl" value="http://ecvaawknys1/sites/FPS/" />
</appSettings>
```

*Kodexempel 8. Hämtningen av mallarna som läses in i en Byte Array.*

```
if (language == "Swedish")
{
    templatePath = ConfigurationManager.AppSettings["TemplateSwedish"];
    bytes = File.ReadAllBytes(templatePath);
}
```

För att få mallen att visas i Richedit-kontrollen var man tvungen att konvertera Byte-arrayn till en MemoryStream före det gick att ladda dokumentet i textredigeraren.

Det finns tre olika mallar av offerten: svenska, finska och engelska. I mallen finns det information om priser, rabatter, leveransvillkor, kontaktpersoner, produkter osv., med andra ord det som behöver finnas i en offert. Mallen är skapad så att all information i dokumentet, som kommer att skilja sig från offert till offert, visas som fält med hårda klamrar runt namnet, se figur 13 för exempel. Detta är ett enkelt sätt för att lägga in data från en databas till dokumentet. Då är det lätt att se var i dokumentet som data skall infogas. Några exempel på data som ändras är datum, företag, adress etc.

	<b>Offert</b> [Date]
[Company] [Contact] [Address] [Postalcode] [City]	
<b>Anbud råmaterial Ostra hamm</b>	Vi tackar för er förfrågan och offererar våra produkter enligt bilaga Priserna gäller till helhetsleverans.
<b>Priser</b>	Netto moms 0 %. Vid fakturering tillkommer gällande mervärdesskatt (23 %).
<b>Rabatter</b>	Rabatterna räknas ifrån ikraftvarande prislista. För tillfället gäller vår prislista daterad 1.2.2012 (finns på vår hemsida <a href="http://www.kwhpipe.fi">www.kwhpipe.fi</a> ).
<b>Prisgiltighet</b>	Priserna är fasta för leveranser t.o.m. 31.5.2012. Efter detta datum är priserna bundna till råmaterialpriset (ICIS <a href="http://www.chemicalbusiness.com">Chemical Business</a> ). Fr o.m. anmälningsdatumet överförs 70 % av råmaterialprishöjningen på produkternas priser. Om råmaterialpriset stiger mindre än 5 % höjs inte priserna. Indexuppföljningen av råmaterialpriset börjar 1.6.2012 Jämförelseindex är mars 2012
<b>Leveranstid</b>	Enligt överenskommelsen.
<b>Leveransvillkor</b>	DAP Lemland, Incoterms 2010. Köparen ansvarar för lossningen av lasten.
<b>Fakturering</b>	14 dagar netto
<b>Övriga villkor</b>	Enligt våra allmänna försäljningsvillkor ( <a href="http://www.kwhpipe.fi">www.kwhpipe.fi</a> ).
<b>Giltighet</b>	Vår offert är i kraft 60 dagar från 22.3.2012
<b>Kontaktpersoner</b>	[TechSupportContact] [KWHPipeContact]
	Kontakta oss gärna för ytterligare information.

Figur 13. En del av offertutskriftsmallen.

Produkterna skulle visas i en bilaga till offerten. Problemet med mallen var hur den skulle designas för att kunna få in en lista med produkter från en datakälla till mallen. Att få in data var inget problem, men det var svårt att få det att se bra ut. Det skapades en tabell på en ny sida i Word-dokumentet som innehåller två rader, en med rubrikerna på kolumnerna och en andra som är tom. Då kunde man i koden söka igenom texten i RichEdit-kontrollen efter tabellen och fylla ut rader i tabellen enligt antal produkter för ordern. I figur 14 kan man se hur produkttabellen ser ut.



The screenshot shows a software window titled "Offert" with a ribbon interface. The "File" tab is selected, showing standard file operations. On the left, a language menu is visible with "English", "Finnish", and "Swedish" options. The main area contains a table titled "Prisbilaga" with the following data:

Produkt	Mängd	eur/enh
WehoAntistatic PN10 90/77	5,0 m	30,00 €
WehoAntistatic PN10 250/215	6,5 m	837,00 €
- Mechanical Couplings/Vicalic	6 pc	
- KWH Engineer support	1 pc	
		<hr/> 867,00 €

Figur 14. Screenshot av produkttabellen i offerten.

Ett annat problem som uppstod var att det fanns möjlighet att utelämna data i vissa fält. Till exempel behövde man inte fylla i fältet TechSupportContact i kontaktpersoner. I offertmallen fanns det dock rader för alla kontakter som det gick att fylla i information om, så därför var man tvungen att lämna en tom rad i offerten ifall inte info för kontakten fanns tillgänglig.

Under Kontaktpersoner har telefonnummer, epostadresser och andra kontaktuppgifter för kontaktpersonerna lämnats bort. Slutsatsen drogs att det är lättast att det fylls i manuellt före offerten sparas manuellt. Detta främst för att det var svårt att få det att se bra ut när man försöker få in informationen dit.

## 4 Resultat

Resultatet av arbetet är en modul som skapar en offertutskrift på basen av valda produkter och installationskostnader för en offert. Offertutskrifter kan skapas på tre olika språk: svenska, finska och engelska. Om nya språk tillkommer i framtiden är det inte något stort arbete att få med dem i programmet. Modulen är en del av en Sharepoint-applikation med webparts. Programmet är just nu under testning och små ändringar har skett medan detta examensarbete skrivits och kommer säkerligen också att ske i framtiden. Applikationen kommer att användas internt på KWH Pipe av anställda för att skapa offerter åt KWH Pipes kunder.

## 5 Diskussion

Eftersom Ecraft ägde licenser till Devexpress-verktygen, var det ganska klart att lösningen skulle göras med hjälp av dessa. Deras kontroller har en snygg design och mycket bra funktioner. Dessutom görs många av lösningarna på Ecraft med hjälp av Devexpress produkter. Det fanns en textredigerare för WPF som också kunde ha använts, genom att visa en WPF-sida i popup-fönstret i stället för en Silverlight-sida.

Silverlight-versionen av textredigeraren valdes främst p.g.a. att det på Devexpress-hemsidan fanns en bra demo för hur kontrollen fungerar. Det medförde att man fick en bra bild av hur man kunde fortsätta.

När det kommer till användargränssnittet för Silverlight-sidan som visas, så kunde jag ha visat listan för olika språkval på ett annat sätt. Som det är nu så visas tillgängliga språk i en "listbox" till vänster om texteditorn. En möjlighet var att bygga in det i verktygsfältet för textredigeraren, på samma sätt om man t.ex. byter font så skulle man byta språk genom att välja i en dropdown-lista. Men det som är bra med nuvarande lösning är att det är tydligt vilka språk som finns att välja mellan.

För att få in data i rapporten fanns det också olika möjligheter. Det som skulle vara smidigast var en Mail-Merge-funktion som fungerade så att man specificerar en datakälla till textredigeraren och därifrån får in data. För detta hittades dock inte en lösning för

rapporthantering. Resultatet blev att fält läggs in i rapportmallen och när data ska in i dokumentet så söker man efter fältet och byter ut ordet mot önskad information.

Detta var en av de första uppgifterna jag fick som anställd på Ecraft och med brist på arbetserfarenhet var det en utmaning. Jag var tvungen att söka efter fakta med jämna mellanrum och fick en hel del hjälp av andra anställda på Ecraft för att resultatet skulle bli bästa möjliga. Eftersom många av de Silverlight-kontroller jag använt är från Devexpress, har jag funnit mycket nödvändig information på deras hemsida där det funnits demoversioner av kontrollerna så att man sett vilka funktioner som finns och hur de fungerar. Jag hade ingen tidigare erfarenhet av att arbeta med Silverlight, Sharepoint och WCF, så jag har lärt mig mycket medan arbetet gjordes.

Jag har märkt att den erfarenhet jag fått av att göra examensarbetet ger en bra grund för arbetslivet.

## 6 Källförteckning

- About Us (u.å.). [www.Devexpress.com](http://www.Devexpress.com) (Hämtat: 20.6.2012).
- Bakharia, A. (2002). *C#: fast&easy web development*. Premier Press.
- Dayley, B. (2008). *Silverlight 2*. Wiley Publishing Inc.
- Företagspresentation (u.å.). [intra.ecraft.com](http://intra.ecraft.com) (Hämtat: 12.7.2012).
- Introduction to Building Windows Communication Foundation Services (u.å.). [msdn.microsoft.com/en-us/library/aa480190.aspx](http://msdn.microsoft.com/en-us/library/aa480190.aspx) (Hämtat: 21.6.2012).
- Klein, S. (2008). *Professional LINQ*. Wiley Publishing, Inc.
- Liu, M. (2010). *Wcf 4.0 Multi-Tier Services Development With Linq To Entities*. Packt Publishing
- Magennis, T. (2010). *LINQ to objects using C# 4.0*. Addison-Wesley
- Mayo, S. (2010) *Microsoft Visual Studio 2010:0 A Beginners Guide*. Mc Graw Hill.
- Microsoft Visual Studio* (u.å.). [en.wikipedia.org/wiki/Visual\\_Studio](http://en.wikipedia.org/wiki/Visual_Studio) (Hämtat: 26.7.2012).
- Rader, D., Hinkson, G. & Beres, J. (2008) *Silverlight 1.0*. Wiley Publishing Inc.
- Skansholm, A. (2008). *Skarp programmering med C#*. Studentlitteratur.
- Snow, A. (2009). *Game programming with Silverlight*. Course Technology.
- Troelsen, A. (2012). *Pro C# 5.0 and the .NET 4.5 Framework*. Apress.