Denis Blazevic | Programmering 2 | May 19, 2016

Hotelsystem - Rapport

Windows forms-baserat hotelsystem



# Inledning

Projektet är menat att visa upp att man har de kunskaper för att skapa ett program som har skild GUI och program kod, klasser samt skriv- och läsfunktion när det kommer till filer.

Jag valde att göra om webbserverprogrammeringens projekt till en Windows-Forms applikation, där personal kan boka rum, ta bort bokningar och kolla tillgängliga rum.

# Genomförande

Det första som gjordes var en enkel projektplan som blev relativt kort då detta projekt handlar om att konvertera från ASP.NET till WinForms-applikation. Efter att projektplanen gjordes startades programmeringen igång i WPF, men byttes snabbt tillbaka till WinForms eftersom jag inte tyckte om GUI som jag gjorde och hade ingen tid att göra den bra. WinForms användes till slut eftersom det är enklast när det gäller att skapa GUI men också för att vissa servrar funktioner inte finns i WPF men som finns i WinForms som enkla ”relativa connectionstrings”.   
Efter att GUI var klart, sattes programmeringen i gång. Jag delade upp programmet i tre huvuddelar – databashanterare, användarhanterare och huvudprogrammet där GUI anropar dessa hanterare för att göra saker som t.ex. lägga till nya bokningar eller för att ta bort bokningar.   
Databashanterare fick jag problem med eftersom jag använde mig utav tekniken ”LINQ to SQL” som gör det väldigt enkelt att arbeta mot en databas med hjälp av enbart C# kod. Inga ”string query” behövs och inga manuella databas anslutningar. Med LINQ to SQL kan man skapa automatiska databasmodeller som fungerar direkt mot C# kod utan att behöva ställa in många saker. Problemet som jag stötte på var att jag inte läste igenom MSDN guiden ordentligt, och missade därför att skapa databasmodellen automatiskt istället för manuellt (manuellt går men är inte rekommenderat.).   
Användarhanteraren fungerar på så sätt att nya användare kan skapas (dock inte inlagt i GUI) och varje gång en användare skapas sparas användarnamnet i databasen, lösenordet hashas först och sedan sparar i databasen. För att logga in en användare kollar applikationen om lösenordet de skriver in, ger tillbaka en hash som är likadan som den i databasen. Om det stämmer så loggas användaren in.

# Utvärdering

Projekten tog längre tid än vad jag trodde pga. att jag missade att skapa databasmodellen automatiskt. Annars flöt det på och jag lärde mig nya saker som t.ex. användarhantering utan färdiga bibliotek/metoder (som ASP.NET Identity) men även LINQ to SQL.

# Slutsats

Applikationen fungerar så som tänkt. Tyvärr är inte några funktioner inlagda pga. brist av tid, dessa funktioner är följande – personal kan inte lägga till nya rum i databasen och personal kan inte skapa nya konton. Annars fungerar applikationen som tänkt.