# Individuellt programmeringsprojekt

Inledning

Detta projekt handlar om att skapa ett bibliotekssystem som gör det möjligt för användare att låna och hämta böcker, samt så kan administratörer hantera bokdatabasernas innehåll. Applikationen ska ge användarna möjlighet att registrera sig, logga in och söka efter böcker baserat på olika kriterier som titel, författare och ISBN. Administratörer kommer att ha extra funktionalitet för att uppdatera, lägga till och ta bort böcker från systemet.

Systemet ska även involvera en funktionalitet som möjliggör för administratörer att förhindra så att böcker inte blir dubbelbokade, så systemet ska hålla koll på bokens tillgänglighet.

Planering och Tidsplanering

**1. Planering och Strukturering (1 november - 5 november)**

* **Uppgifter:**
  + Skapade en WPF-applikation med XML funktionalitet.

En bild som visar text, skärmbild, dator, skärm

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, programvara, dator

Automatiskt genererad beskrivning

Bild 2: Bilden ovan visar koden för det nya projektet.

Bild 1: Skapar ett WPF-projekt, här ser vi fönstret.

**2. Användarhantering (6 november - 10 november)**

* **Uppgifter:**
  + **En bild som visar text, skärmbild, programvara, dator

    Automatiskt genererad beskrivningEn bild som visar text, programvara, skärmbild, dator

    Automatiskt genererad beskrivning**Implementerade knappar för registrering och inloggning för **Användare** och **Administratörer.**

Bild 3: I denna bild så lägger jag in knappar för inloggning samt lägg till och radera böcker

.

Bild 4: Bilden visar kod för knapparna på inloggningssidan.

**En bild som visar skärmbild, text, programvara, Multimedieprogram

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, skärm, programvara

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, design

Automatiskt genererad beskrivning**

Bild 8: Bilden visar koden som ger knapparna deras form, de två röd-orangea pilarna pekar på två olika stylings som sen tillämpas på varje element om ska stylas.

Bild 5: Här har jag skapat tre separata sidor, en som sköter inloggningen och de andra två är User och AdminPage som ska få olika funktionaliteter.

Bild 6: Bilden visar den uppdaterade inloggnigssidan där knapparna aktiveras om lösenordet stämmer.

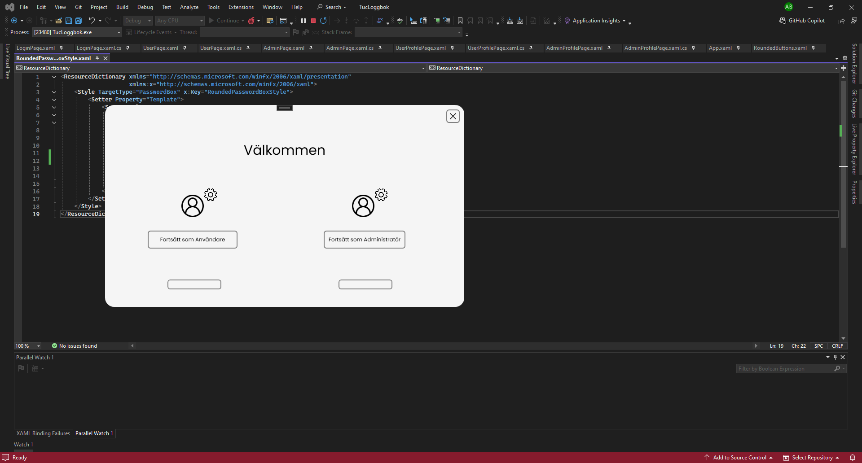


Bild 7: Här är en uppdaterad och mer stilren layout på inloggnigsskärmen

**3. Bokhantering och Lånehantering (13 november - 17 november)**

* **Uppgifter:**
  + En bild som visar text, skärmbild, Teckensnitt, dokument

    Automatiskt genererad beskrivningEn bild som visar text, skärmbild, dator, programvara

    Automatiskt genererad beskrivningImplementerade ett system för att hantera böcker och lånehantering.

Bild 9: Bilden visar ett fel- meddelande vilket gjorde att programmet skickades till TryCatch-blocket.

Bild 10: Här visar bilden en lyckad implementering av lånehistoriken.

**4. Gränssnitt och Sökfunktionalitet (20 november - 24 november)**

* **Uppgifter:**
  + Designade gränssnitt och sökfunktioner.

Bild 11: I bilden ovan så lades en registreringsknapp till så användare kan registrera sig och får välja vilken typ deras konto är, samt välja lösenordet.

En bild som visar text, skärmbild, programvara, Multimedieprogram

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, programvara, Multimedieprogram

Automatiskt genererad beskrivning

Bild 12: I bilden så har jag implementerat en funktion där användaren får ett felmeddelande om fel kontotyp används för att logga in.

**En bild som visar text, skärmbild, programvara, multimedia

Automatiskt genererad beskrivning**

Bild 13: Bilden visar en ny design på fönstren där titeln visas för att tydliggöra fönstret man är i.

**5. Testning och Dokumentation (27 november - 3 december)**

* **Uppgifter:**
  + Jag testar den kompletta applikationen nu för att se om allting fungerar som det ska, och alla funktioner samt hur jag tänkte ska nu dokumenteras.

Arbetsprocess (Loggbok)

* **Dessa idéer fick jag**:

Jag fick först en idé om att skapa ett par knappar för att lägga till böcker, radera, spara logg, logga in som admin/användare (dessa två sistnämnda knappar är från början inaktiverade), jag har även två textbox som aktiverar knapparna (om rätt användarnamn/lösenord matas in) för att fortsätta antingen som admin/användare, när dessa knappar är aktiverade och klickas ned så visas ett popup-meddelande som meddelar användare/admin om lyckad inloggning.

Jag fick senare ett axplock av idéer kring knapparna och vad de olika knapparna skulle göra. Användaren omdirigeras till en individuell UserPage där det finns ett par funktionaliteter som i att se bokloggen, låna och hämta böcker medan administratören får liknande men bredare funktionalitet där man kan hämta, kolla loggen, lägga till böcker, radera böcker, redigera loggen samt radera loggen.

Jag fick en massa idéer vid design av fönstren, knapparna och textboxes, först så tänkte jag att det skulle vara en bra ide att implementera mina egna minimerings -och förstorings knappar uppe vid högra hörnet samt snap-funktionen likt Windows men det tog för mycket av tiden, det fanns redan så jag fokuserade på det viktigaste, som var då användarvänligheten och att det var bättre att implementera funktioner som underlättade för användaren, så jag bestämde mig för att gå tillbaka till att ge titel till fönstret och ikoner som tydliggör fönstret som är aktuellt. Utöver det så kom en rad olika idéer om att skapa funktioner som MessageBox för att ge användaren eller admin information om vad som hände/hänt vid utloppet.  
  
I min kod funderade jag på att lägga till många kommentarer som förklarar vad varje kodsnutt gör, eller bara korta och koncisa kommentarer, eller inga alls, eftersom metodnamnen redan förklarar vad funktionen gör. Efter att fått lite feedback från en duktig lärare, fattade jag ett beslut, metoderna var självförklarande så jag uteslöt att lägga till kommentarer.

**Detta gjorde jag och det tog så här lång tid**:

* **Problem och lösning**

Innehållet (knappar, etiketter och textboxes) ”lämnade” fönstren när de förstorades så för att lösa detta så användes flera <Column.Definitions/> samt <Row.Definitions/>, det tog rätt så lång tid att få allt innehåll att positionera sig inom sina kolumner och rader.

När inloggningssidan skadades så funderade jag på hur jag skulle implementera funktionaliteten med inloggning så jag provade på alla sätt, från manuell inloggning till automatisk generering av lösenord men typiskt så såg jag inga lagringar inne i databasen, jag fick då en annan idé med BCrypt hashning av lösenord som gör det säkrare vid inloggning och som förhindrar intrång, denna lösning fungerade och lösenordet och användarnamnet lagrades inne i databasen, denna lösningen tog dock en rätt så stor del av tiden (ett par dagar) då jag fick läsa på en hel del om den här krypteringsnyckel.

Att lära mig om Querys tog mig rätt så lång tid då man måste komma ihåg i vilken struktur man ska lägga koden i. det verkade sen väldigt logiskt när man hade Query och tabellen vid sidan om så man kunde koppla ihop hur och vad olika saker innebar.

* **Tidsplanering**

Tidsplaneringen tycks vara inom en bra tidsram, allting gick okej tyckte jag, arbetet delades upp med jämna mellanrum.

* **Detta tänkte jag på i arbetet**

Jag fokuserade på att göra gränssnittet mellan användare och applikation så enkelt som möjligt. Om användaren stöter på problem eller hinder så ville jag säkerställa att användaren får tydlig information som förklarar vad felet är, jag tycker detta förbättrar användarupplevelsen genom att eliminera förvirring kring fel som har uppstått. Även om allting inte är felfritt så har jag försökt mitt bästa.

* **Måste något förändras? Vad och varför?**

Planeringen fungerar bra eftersom jag delar upp vad som ska göras så att jag inte ger mig på allting på en och samma gång, detta kan göra så att jag missar någonting eller att jag lägger in fel funktioner på fel ställe.

Det jag har ändrat på relativt mycket var designen på fönstren, jag testade att ta bort titeln från alla fönster men jag inkluderade dem igen och dessutom lade till ikoner för att bättre visualisera för användaren vilken sida de är på.

Funktionell Översikt

**Användarhantering**

1. LoginPage

* Användaren/administratör kan logga in baserat på rollen och lösenorden hashas med BCrypt för ytterligare säkerhet.

1. Adminhantering

* Admin får efter inloggning fler funktioner än Användare, visa, radera loggar, söka efter böcker samt redigera loggar.

**Bokhantering**

1. Lägga till böcker

* Admin kan lägga till böcker genom att fylla i titel, författare, ISBN och status.
* Detta sparas i databasen via en SQL INSERT-fråga.

1. Ta bort en bok

* Admin kan ta bort en bok från databasen genom att välja en bok från listan.
* Detta är möjligt med SQL-frågan DELETE.

1. Söka efter bok
   * + Användare samt admin kan söka efter böcker baserat på titel, författare och ISBN.
     + SQL SELECT-fråga med en LIKE-operator används för att hämta böcker som matchar frågan.
2. Uppdatera loggen
   * + Den här funktionen hämtar alla böcker från databasen.
     + SQL SELECT-frågan göra detta möjligt.

**Lånehantering**

1. Visa lånehistorik
   * Admin kan visa en användares lånehistorik, vem som lånade boken och om den är återlämnad eller inte.

Beslut och Reflektioner

* **Detta beslutade jag/vi**:

Jag tog beslutet om att göra ett enkelt system för användare/administratör och inkludera felmeddelanden så gott det gick för att notifiera om det uppstod fel.

* **Hur gick det med det jag/vi bestämde förra gången?**

Förra gången så missade jag att lämna in gruppens tankar inklusive mina egna om projektet och vad vi gjorde under projektets gång, feedback togs emot och i detta projekt så hoppas jag att allting finns med.

* **Egna tankar**

Det har varit ett riktigt spännande och lärorikt projekt, jag känner att jag har lärt mig en hel del om SQL och hur man ska strukturera koden, det är även väldigt intressant och spännande att se hur fönstren i XAML ändras i realtid, jag tyckte också att det fick mig att vilja förfina det ännu mer.

Något jag fann extra intressant var SQL frågorna, först visste jag inte hur, eller om det ens var möjligt, att en knapp eller textbox kunde kommunicera med databasen när den var i fokus, men det gick över förväntan. Jag tänkte först att man var tvungen att uppdatera databasen och skicka in frågor manuellt men det visade sig vara tidskrävande. Efter att ha läst och lärt mig mer insåg jag att det kunde skötas automatiskt i koden med en string Query, jag ser fram emot att lära mig mer om det här.

Källor och Referenser

1. [The MessageBox - The complete WPF tutorial](https://wpf-tutorial.com/dialogs/the-messagebox/) [Hämtat: 2024-10-28].
2. [(18) Start Building a Desktop App in WPF C# - WPF C# Tutorial Part 1 - YouTube](https://www.youtube.com/watch?v=9Pj5QB_AMtc&list=PLJJcOjd3n1Zegr2CaA78RWF9IIgeyq0xh) [Hämtat 2024-10-29].
3. [WPF C# Professional Modern Flat UI Tutorial](https://www.youtube.com/watch?v=PzP8mw7JUzI) [Hämtat 2024-10-30].
4. [Microsoft Copilot: Din AI-assistent](https://copilot.microsoft.com/) [Hämtat 2024-10-30].
5. [Animation - WPF TUTORIAL](https://www.youtube.com/watch?v=fOIJzFjx9vs) [Hämtat 2024-11-02]
6. [The HAVING statement - Learn SQL - Free Interactive SQL Tutorial](https://www.learnsqlonline.org/en/The_HAVING_statement) [Hämtat 2024-11-03]
7. [Learn SQL - Free Interactive SQL Tutorial](https://www.learnsqlonline.org/) [Hämtat 2024-11-04].
8. [SQL Easy: The Best Way to Learn SQL (It's FREE!)](https://www.sql-easy.com/select_columns) [Hämtat 2024-11-04].
9. [SQL Join Query With LINQ](https://www.c-sharpcorner.com/article/sql-join-query-with-linq/) [Hämtat 2024-11-05].
10. [MySQL Functions](https://www.w3schools.com/sql/sql_ref_mysql.asp) [Hämtat 2024-11-11].
11. [MySqlDataReader.GetDateTime, MySql.Data.MySqlClient C# (CSharp) Code Examples - HotExamples](https://csharp.hotexamples.com/examples/MySql.Data.MySqlClient/MySqlDataReader/GetDateTime/php-mysqldatareader-getdatetime-method-examples.html) [Hämtat 2024-11-11].
12. [.NET 6.0 - Hash and Verify Passwords with BCrypt | Jason Watmore's Blog](https://jasonwatmore.com/post/2022/01/16/net-6-hash-and-verify-passwords-with-bcrypt) [Hämtat 2024-11-17].
13. [Use BCrypt to Hash Your Passwords: Example for C# and SQL Server- CodeProject](https://www.codeproject.com/articles/475262/use-bcrypt-to-hash-your-passwords-example-for-csha) [Hämtat 2024-11-17].
14. [Bcrypt & Password Security - An Introduction](https://www.youtube.com/watch?v=O6cmuiTBZVs&t=20s) [Hämtat 2024-11-21].
15. [bcrypt - Wikipedia](https://en.wikipedia.org/wiki/Bcrypt) [Hämtat 2024-11-21].
16. [What is the difference between a StackPanel and DockPanel in WPF? - Stack Overflow](https://stackoverflow.com/questions/4320861/what-is-the-difference-between-a-stackpanel-and-dockpanel-in-wpf) [Hämtad 2024-11-28]

Förberedelser och Nästa Steg

* **Vad ska förberedas till nästa gång?**

Jag måste förbättra mig inom Debugging och kolla steg för steg i locals fönstret vad som händer i koden, debugging gör jag men inte så ofta som jag borde och där måste jag se till att förbättra mig.

* **Vad ska jag ta upp med handledaren/läraren?**

Be om feedback om min kodstruktur och var jag kan förbättra mig, jag ska diskutera vad som är bra och de hinder som uppstod samt mina lösningar. Jag ska även ta upp om det finns något som jag kunde ha implementerat som jag inte tänkt på.