

GreenThumb

Scenario

GreenThumb är ett startup som specialiserar sig på att föra trädgårdsodling närmare människor i urbana områden genom sin innovativa app. Med visionen att göra trädgårdsskötsel tillgänglig och engagerande för alla, har GreenThumb anlitat dig, en systemutvecklare för att bygga appen.

Berörda kursmål

- ER-modellering
- Databasdesign
- Användning av SQL
- Kryptering och säkerhet i databaser
- Transaktioner
- ORM (Entity Framework)
- Effektiv databasdesign
- Använda databashanterare
- Konstruktion och hantering av kryptering
- Prestandaoptimering
- Lagring och hantering av information i databaser

Förutsättningar

- Individuellt arbete.
- Kommentarer krävs endast där du själv bedömer det nödvändigt.
- Utgå från en WPF-applikation i Visual Studio 2022.
- Använd SQL Server och code first eller database first approach med EF Core och .NET.
- Följ det tillhandahållna ER-diagrammet, och inkludera eventuella ytterligare entities i detta efter behov.

- Använd Repository-pattern och/eller Unit of Work-pattern.
- Seed:a databasen med information om växter och skötselråd.
- Lösenord ska vara krypterat i databasen.

Struktur

PlantWindow

- Se och söka på växter.
- Sökfunktion för växtnamn.
- Knapp för att komma till AddPlantWindow.
- Knappar "Details" och "Delete" för att hantera valda växter i listan, med tillhörande varningsmeddelanden.
 - Om man tar bort en växt ska även tillhörande skötselråd försvinna från databasen.

AddPlantWindow

- Lägga till ny växtinformation.
- Fält för växtens namn, skötselråd och planteringsdatum.
- Knapp för att spara den nya växten.
- Varningsmeddelande om informationen inte är komplett.
- Om man försöker lägga till en växt som redan finns i databasen ska ett varningsmeddelande dyka upp.

PlantDetailsWindow

- Visa och ändra detaljer om en växt.
- Låsta inputrutor för växtinformation.
- Knappar för att låsa upp inputrutor för att möjliggöra ändringar.
- En knapp för att spara ändringar (namn, skötselråd och planteringsdatum).
- Knapp för att lägga till växten i sin trädgård (se MyGardenWindow nedan).

MyGardenWindow

- Visa användarens trädgård och alla växter i den.

- Visa och arrangera växter i en virtuell trädgård.

SignInWindow

- Logga in med användarnamn och lösenord.

RegisterWindow

- Registrera en ny användare.
- Validering för att förhindra duplicering av användarnamn.

Bedömning

Godkänt

- Välstrukturerad och buggfri kod.
- Uppfyllande av alla "Förutsättningar".
- Implementering av alla funktioner i "Struktur", förutom det med understruken text (VG-krav) och det med *kursiv text* (nice-to-have).

Väl Godkänt

- Uppfyllande av "Godkänt"-kraven.
- Implementering av all funktionalitet med understruken text i "Förutsättningar" och "Struktur".
- **Reflekterande Projektanalys:** Inkludera en välformulerad och insiktsfull projektanalys i din projektinlämning. Denna analys bör vara på cirka en A4-sida och innehålla en reflektion över projektets arkitektur, valda tekniska lösningar, och de utmaningar och lärdomar du stött på under projektets gång. Den ska inte bara beskriva vad som gjordes, utan också varför vissa beslut togs, hur de påverkade projektets utfall, och vilka insikter du fått som en framtida systemutvecklare. Denna analys är din chans att demonstrera din förmåga att inte bara utföra tekniskt arbete, men också att reflektera över och lära av processen.

Nice-to-have

- All *kursiv text* i avsnittet "Struktur" är extrafunktioner att implementera om tid finns.

Tips

Här är några tips för att framgångsrikt hantera och genomföra projektet:

- **Förstå uppgiftens krav:** Börja med att noggrant läsa igenom uppgiftens specifikationer och se till att du förstår varje aspekt av vad som krävs. Skissa gärna upp klasser och entiteter med penna och papper.
- **Planera noggrant:** Skapa en projektplan som inkluderar en tidsuppskattning för varje del av projektet. Detta hjälper dig att hålla dig på spåret och säkerställer att alla delar av projektet blir färdiga i tid.
- **Utveckla och testa funktioner systematiskt:** Testa varje transaktion med databasen noggrant för att säkerställa att de fungerar som förväntat. Börja med att testa alla CRUD (create/read/update/delete) funktioner i SSMS med SQL innan du börjar koda för mycket! Implementera sedan Repository pattern för att undersöka att alla transaktioner fungerar som de ska. **Ett viktigt råd är att INTE börja med att skapa ett raffinerat användargränssnitt (UI), utan börja med att få alla transaktioner med databasen att funka som det ska.**