C# kursen att ha med i dokumentationen / projektet

Användning av **overloads** är när samma metod används flera gånger med olika parametrar
I Vissa fall kan man minska koden med default parametrar.

Factory pattern, skapa en grund, använd overloads på methoder.

Service pattern, är det bara att följa SOLID reglerna fast i services? Vad saknas? Repository pattern,

Interfaces med polymorph och decoupling

Multicast delegates och delegates användning vid eventtriggers - vilka eventtriggers finns det ?

WPF - Xaml, grid system, med styling och komponeneter att använda och styla, c# koden kopplad med cs filen. Använd funktionellprogrammering, obejkt orienterad programmering och slutligen jobba med delegate event triggers.

Klassbibliotek användning? - lägg till alla facotrories, services, och handlers etc, med alla interfaces, overloads, polymorphade och uppdelade funktioner som använder delegates vid eventtriggers samt dtos, (kollar mer på dtos)

Strukutrer mappaen enligt domain driven data, business > helpers/services och i domain > Dtos, factories, models., vi behöver även data så har vi logiken där.

Eventuell mapp struktur.
src/ ├── Presentation/ │ ├── Views/ │ │ ├── MainWindow.xaml │ │ ├── UserControl.xaml │ ├── ViewModels/ │ │ ├── MainViewModel.cs │ │ ├── UserViewModel.cs │ ├── Commands/ │ │ ├── RelayCommand.cs ├── Domain/ │ ├── Models/ │ │ ├── User.cs │ │ ├── Order.cs │ ├── Factories/ │ │ ├── UserFactory.cs ├── Business/ │ ├── Services/ │ │ ├── UserService.cs │ │ ├── OrderService.cs │ ├── Helpers/ │ │ ├── ValidationHelper.cs ├── Data/ │ ├── Repositories/ │ │ ├── UserRepository.cs │ │ ├── OrderRepository.cs │ ├── DbContext.cs ├── Dtos/ │ ├── UserDto.cs │ ├── OrderDto.cs

Använd, interfaces, jobba mellan protected osv, skapa lösenord först sedan menyvalet, lösen ska använda one way ändrandes osv, använd delegates att aktivera från flera crud itnerfaces vanlig, json, User,Contact, Bot typ,

Tänk på att du ska använda abstrakt klass så du kan använda konkreta versioner som amdin ohc normal user detta blir en model version

Försök alltid checka med chatgpt för try/Catch, nullable types et

Sätt alltid null på modellerna, speciellt vid testning så du kan få rätt värde använd inte strings etc

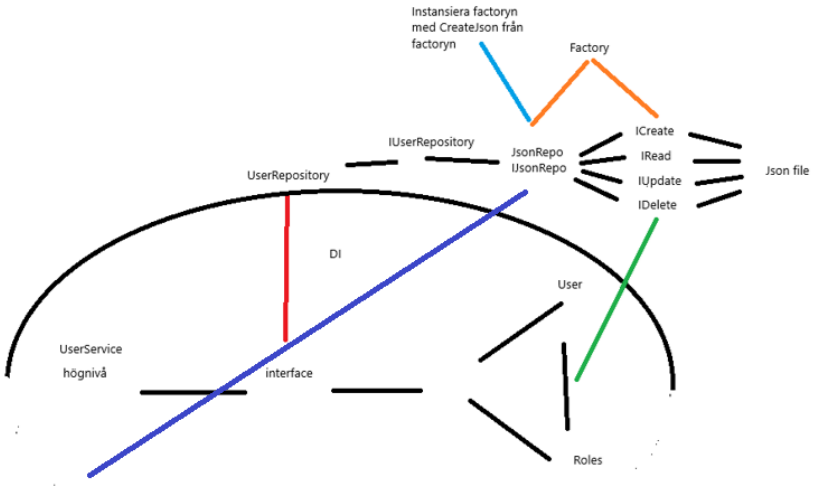
Tänk på att skriva med att XUnit skriver till databas eller använder methoderna i JsoNRepo medans Moq simulerar.

Använd generiska T för typerna i ICretae/IDelete/Update/IRead allltså i de uppdelade interfaces för crud.

Så med generiska typer gör vi generiskt på interfaces och kan kan nog använda samma JsonRepo men för IContactRepository istället för IUserRepository så båda Services använda samma JsonRepo och som man hade kunnat byta till t.ex SqlRepo med mera.

Anvönd abstrakt på User > Admin/Normal men även en base abstrak to UserService och ContactService så vi har en bas klass till crud funktioner **VIKTIGT** tänk också på att när du använder abstrakta klasser att du använder abstrakta metoder för medother som behöver ersättas i varje underklass och att man använder virtuella metoder för de metoder som ska finnas mellan klasserna. Då kan vi använda metoderna previs som om det existerade i klaseen

men aväen polymorpha de metoder som ine får vara samma.



Notiser

Använd t.ex genereteld Helper skapa overloads om du har flera id generatorer

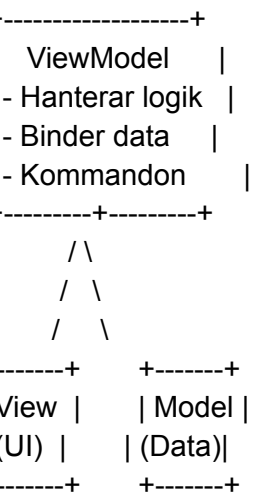
Interface sereration har vi i ICreate / IDelete etc som används av JsonRepository men instansieras av fabriken

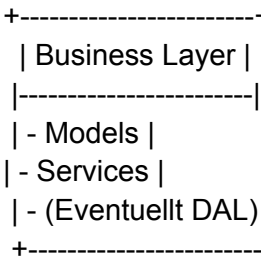
Föratt förklara negativa värden med injektion så UserService tar in UserRepo som Tar in JsoNRepo om vi har en app som kallar en method i userservice som kallar userrepo som kallar jsonrepo som kallar userService igen har vi en cirkel som loopar sönder. Och kan skapa operativa delar i varje method eller i json repo, så vi kan skapa 10000 users i ett eller loopa sönder datorn i en aktitektur som inte bara är en for loop. Utan DI ska användas från app till databas eller datafil eller liknanade, eller slut del iaf t.ex event, eller hämtning av data.

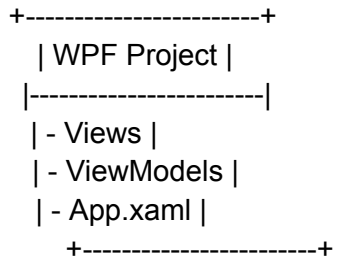
Efter 16 dec
Mocka klart systemet med alla metoder för basic testing.
Läs Mvvm och Maui/Blazor evetuellt även WPF

Assembly beskrivningar är meta taggar, lite som meta taggar i html eller exakt samma.
Programmet körs via App classen och Xaml sidorna visas

Mvvm







18 jul handla, dammsaugnign och julgran uppsättning läsa videos
19, Tvättning läsa videos
20 julmarknad runt lund läsa videos
21 Programmera WPF
22 Programmera WPF
23 Bingolotto kväll Börja läsa videos som öppnat
24 jul
25 andra jul

26 läsa
27 läs
28 läs/börja programmera maui/blazor
29 programmer maui blazor
30 Handla inför nyår(erbjudanden kan hittas innan) progammera maui blazor
31 nyår
1 nyår
2 nyår

Kontrollera uppgifter, WPF, Business, Maui/Blazor och Mocking så allt funkar och är somdet ska. Fram till 10 januari alltå 7 dagar

Maui, tittar mer på color resources osv för xaml så vi kan separera stylingen lite från huvudfilen.

Slutspurten
Skapa en Maui project, kolla navigeringsformerna och välj en? Vilken är passande min app? Maui appen ska vara responsiv tänker jag?

Använda Mvvm på maui appen

Tittar hur hanteringen av listor/events funakr

Fixa och kontrollera mockingen så vi verklgen testar alla methoderna i servicesen.

Uppdatera grafiska designen på xaml koden i gamla projektet

Skriv dokumentation/beskrivning och kommentera all kod igen.

Maui kör med Shell navigation och mvvm,

03/01-2025
Teoretiskt: titta klart iallafall halva videon om maui/mvvm
Praktiskt: börja checka av mocking/buttons/appicon på xaml/mvvm versionen.

04/01-2025,
Teoretiskt: Kolla klart videosen
Praktiskt: Börja bygg maui appen med Shell/Mvvm, kontrollera i alla Operativ system.

05/01-2025 - 10/01-2025 Fortsätta bygga på Maui appen, utveckla nya saker om man hittar det med mera. Eventuellt det med listor etc.

Tänk på att innan Maui appen är klar ska det skrivas kommentarer på allt och sedan ska vi skriva en mindre dokumentation.

Fokusera istället vanliga appen/kommentarer och dokumentation.

Kan man bygga en frontend/backend i appen t.ex UserService är kvar i appen men userrepository eller JsonRepository är på servern sem kopplar man med sträng imellan? Tänker dt iaf.

07/01 Dokumentation kvar, Klassdiagram, Text fil i varje Projectfil som summary?