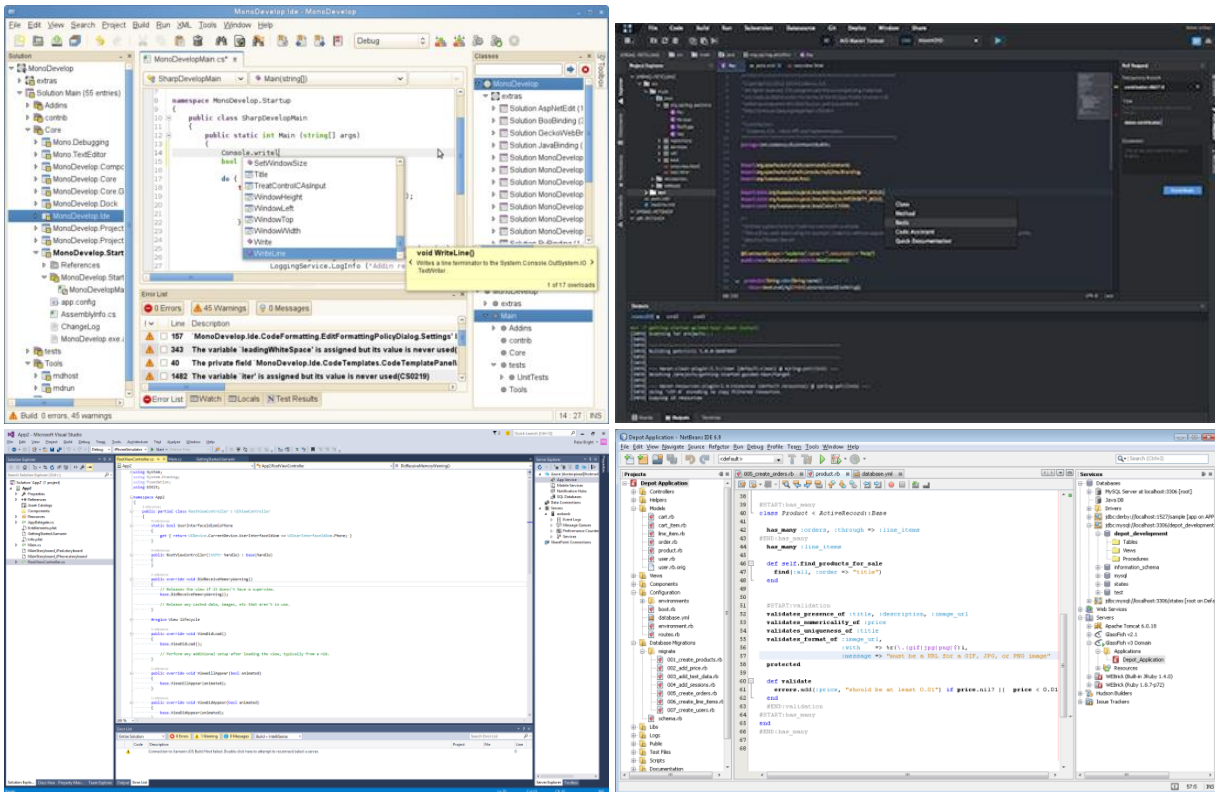


Razvojno okolje

V tehnologiji WPF boste oblikovali grafični uporabniški vmesnik za integrirano razvojno okolje namenjeno razvoju programske opreme. Začnite s pregledom grafičnih vmesnikov že obstoječih aplikacij. Spodaj je nekaj primerov sodobnih razvojnih okolij. Večina vsebuje dodatne funkcionalnosti, mi pa se bomo osredotočili le na najpomembnejše.



Prikazane aplikacije imajo dokaj podobno strukturo. Na vrhu vedno najdemo meni s ključnimi možnostmi. Na levi ali desni najdemo prikaz strukture projekta v drevesni obliki, na dnu aplikacije pa je del za izpis in seznam napak. Osrednji del je namenjen prikazu programske kode. Tekom vaj boste implementirali celovit grafični uporabniški vmesnik za razvojno okolje.

Vaja 1: XAML

obvezno

Vaša naloga je oblikovanje grafičnega uporabniškega vmesnika v XAML z uporabo vgrajenih gradnikov knjižnice WPF. Za zasnovo lahko vzamete katerega koli od zgoraj prikazanih programov ali kakšnega po lastni izbiri.

1. Glavno okno programa razdelite z upravljalci postavitve (angl. Layout Manager), ki smo jih spoznali na vajah (obvezno uporabite vsaj dva različna gradnika). V razporejen prostor vmesnika umestite naslednje gradnike (zgledujte se po obstoječih programih):
 - Menu za prikaz glavnega menija aplikacije,
 - ToolBar za gumbe, ki omogočajo manipuliranje strukture projekta,
 - TreeView za prikaz strukture projekta,
 - ListView za prikaz razrednih metod in spremenljivk,
 - RichTextBox za prikaz besedila (kode),
 - StatusBar za prikaz statusnih besedil.
2. V glavni meni aplikacije (gradnik Menu) vstavite 3 podmenije (gradnik MenuItem): »Datoteka«, »Pogled« in »Orodja«. V prvi podmeni »Datoteka« vstavite postavki »Ustvari projekt...« in »Izhod« (gradnik MenuItem).
3. Gradniku TreeView za prikaz strukture projekta vstavite nekaj vnosov in pod-vnosov, ki predstavljajo datoteke projekta (zgledujte se po obstoječih programih npr. Visual Studio).
4. Gradniku ListView za prikaz objektov dodajte nekaj postavk, ki predstavljajo razredne metode v vašem projektu.
5. Gradniku ToolBar vstavite gumba (angl. Button) za dodajanje in brisanje datotek iz strukture projekta.
6. Gradniku StatusBar dodajte vnos (gradnik StatusBarItem) za prikaz besedil o statusu.
7. Izgled gradnikov naj ostane privzet. Gradnike ToolBar, TreeView in ListView postavite skupaj na smiselno mesto v aplikacij. Gradniki v oknu se morajo smiselno prilagajati velikosti okna (za primer si oglejte obstoječe aplikacije in jim spremenite velikost okna).

Vse vnose v gradnikih si izmislite in naj bodo statični (se ne spreminjajo).

Vaja 2: Osnove C#

4t

Nadaljujmo z delom na aplikaciji razvojno okolje. Tokrat bomo aplikaciji dodali nekaj funkcionalnosti s pomočjo programskega jezika C#. V aplikaciji implementirajte naslednje funkcionalnosti:

1. Glavnemu meniju aplikacije dodajte dogodek izhod, ki zapre program.
2. Glavnemu meniju aplikacije dodajte dogodek ustvari projekt, ki bo ustvaril projekt tako, da bo gradnik TreeView napolnil s strukturo vašega projekta.
3. Glavnemu meniju aplikacije dodajte dogodek zapri projekt, ki zapre projekt, ob tem pa vpraša uporabnika če hoče shraniti trenutno odprti projekt.
4. Gradniku TreeView dodajte dogodek, ki ob izbiri razreda v gradniku ListView izpiše vsebovane metode.
5. Implementirajte funkcionalnost dodajanja nove datoteke v projekt ob kliku na gumb dodaj. Vsebina, ki se doda, naj bo statično določena v C# kodi.
6. Implementirajte funkcionalnost odstranjevanja datoteke iz projekta. Ob kliku na gumb odstrani se naj trenutno izbrana vsebina v gradniku TreeView odstrani. V kolikor ni izbrane vsebine, o tem obvestite uporabnika ali onemogočite gumb.
7. Gradniku ListView dodajte dogodek, ki vam ob izbiri določene metode v gradniku TextBox izpiše njegovo vsebino (kodo). Vsebina naj bo statično določena, vsaka metoda pa mora imeti svojo kodo.
8. Dodelajte izgled vaše aplikacije. Opremite aplikacijo z ikonami, ki jih dodajte vsem gradnikom Button ter vnosom v gradnikih ListView in TreeView (glede na vrsto datoteke). Ikone lahko najdete na spletni strani [IconFinder](#) ali kjerkoli drugje na spletu.

Odpravite napake iz prejšnje vaje, kot so nepravilnosti pri razvrstitvi, napake pri prilagajanju velikosti okna, odmiki med gradniki in poskrbite za lep in smiseln vmesnik.

Vaja 3: Okna

6t

Aplikacijo razvojno okolje nadgradite z novimi funkcionalnostmi, okni in izgledom:

1. Aplikaciji dodajte okno (angl. Window) za nastavitve. Oknu onemogočite spreminjanje velikosti. V oknu uporabite gradnik TabControl za prikaz nastavitvev po sklopih. Dodajte zavihek za nastavljanje programskih jezikov. Implementirajte dodajanje, brisanje in urejanje programskih jezikov ter njihovih tipov aplikacij (glej Visual Studio). Dodajte še sklop (zavihek) za nastavljanje ogrodij (angl. Framework). Implementirajte dodajanje, brisanje in urejanje različnih ogrodij. Podatki, ki se nastavljajo v tem oknu, se morajo shranjevati v nastavitve aplikacije (angl. Settings). Okno odprite ob kliku na postavko »Nastavitve« v podmeniju »Orodja« glavnega menija.
2. Aplikaciji dodajte novo okno, namenjeno ustvarjanju projektov. Okno mora omogočati vnos ali izbiro vsaj šestih podatkov, od katerih so obvezni naslednji: izbira programskega jezika (se mora ujemati z vnesenimi jeziki v nastavitvah, glej točko 1), izbira različnih tipov projektov (se spreminja glede na izbiro programskega jezika), izbira ogrodja (se mora ujemati z ogrodji v oknu nastavitve) in lokacijo na disku. Za ostale podatke se odločite sami. Okno mora vsebovati vsaj naslednje gradnike:
 - TextBox za vnos in prikaz podatkov,
 - ComboBox za izbiro programskega jezika, tipa projekta in ogrodja,
 - Button za potrditev in preklic.
3. Okno iz druge točke dopolnite z dogodkom nastavljanja lokacije na disku. Dogodek naj odpre okno za shranjevanje datotek (angl. SaveFileDialog). Oknu za shranjevanje datotek nastavite filter za XML datoteke. Pot morate prikazati v oknu.
4. Implementirajte funkcionalnost ustvarjanja novega projekta. Ob kliku na postavko »Ustvari projekt...« v glavnem meniju odprite okno iz točke 2.
5. Dodelajte okni iz točk 1 in 2 z oblikovanjem, ki se mora skladati s preostankom aplikacije.

Odpravite napake iz prejšnje vaje in poskrbite, da bodo vse funkcionalnosti delovale brezhibno. Poskrbite za izgled!

Vaja 4: Podatki in XML

7t

Tokrat bomo naši aplikaciji dodali podporo za branje in zapisovanje v datoteke XML:

1. Načrtujte in implementirajte podatkovno strukturo za projekt. Podatkovna struktura mora omogočati shrambo projektnih podatkov in njegove strukture (datotek). Projekt mora biti predstavljen z vsaj šestimi podatki, med katerimi obvezno vključite njegovo strukturo datotek. Datoteke morajo vsebovati razredne metode in spremenljivke. Pazite na pravilno gnezdenje elementov in smiselnost strukture.
2. Podmeniju »datoteka« dodajte novi postavki "Uvozi..." in "Izvozi..." (gradnik Menuitem) ter ustvarite pripadajoča dogodki. Dogodek "Izvozi..." naj prikaže pogovorno okno za shranjevanje datotek (angl. SaveFileDialog), dogodek "Uvozi..." pa naj prikaže pogovorno okno za odpiranje datotek (angl. OpenFileDialog). Nastavite filter za datoteke XML!
3. Implementirajte funkcionalnosti shranjevanja in nalaganja podatkov iz datotek XML z uporabo serializacije. Nalaganje naj prebere podatke iz datoteke XML v podatkovno strukturo in jih prikaže v grafičnem uporabniškem vmesniku, medtem ko shranjevanje shrani podatke v datoteko XML iz podatkovne strukture. Izvoz in uvoz morata biti kompatibilna (struktura datoteke XML mora biti enaka). Poskrbite za brezhibno delovanje!

Odpravite napake iz prejšnje naloge in poskrbite, da bodo vse funkcionalnosti delovale brezhibno.

Vaja 5: Delo s podatki

6t

Razvojnemu okolju dodelajte seznam razrednih metod:

1. Gradnik TreeView za prikaz strukture projekta povežite s podatkovnim virom preko DataBinding.
2. Gradnik ListView za prikaz seznama razrednih metod in spremenljivk povežite s podatkovnim virom preko DataBinding.
3. Z uporabo DataTemplate izdelajte prikaz (izgled) posameznih razrednih metod in spremenljivk za gradnik ListView. Posamezen vnos mora biti predstavljen z več podatki hkrati.
4. Definirano podatkovno strukturo iz vaje 4 dopolnite tako, da se spremembe podatkov avtomatsko osvežijo v grafičnem uporabniškem vmesniku (uporabite INotifyPropertyChanged).
5. S pomočjo DataTrigger, spremenite ikono glede na metodo oz. spremenljivko (glej Visual Studio, class view) v gradniku ListView.

Odpravite napake iz prejšnjih vaj, poskrbite, da bodo vse funkcionalnosti (tudi iz prejšnjih nalog) delovale brezhibno in uredite videz aplikacije.

Vaja 6: Uporabniški gradnik

8t

Opremite aplikacijo razvojno okolje z lastnim uporabniškim gradnikom za upravljanje s strukturo projekta:

1. Ustvarite nov projekt namenjen uporabniškemu gradniku za upravljanje strukture projekta. Gradnik primerno oblikujte v stilu sodobnih razvojnih okolij (glej Visual Studio). Uporabniški gradnik mora omogočati naslednje funkcionalnosti:
 - prikaz strukture projekta v gradniku TreeView (glej vajo 1);
 - prikaz posameznih metod in spremenljivk v gradniku ListView glede na izbor v gradniku TreeView;
 - funkcionalnosti, ki ste jih implementirali v prejšnjih vajah v zvezi s temi gradniki;
 - nad gradnikom ListView definirajte kontekstni meni (angl. Context Menu), ki omogoča dodajanje, brisanje in urejanje razrednih metod in spremenljivk;
 - ob izbiri določene razredne metode ali spremenljivke se koda naj prikaže v gradniku RichTextBox (vaja 1).
2. Ustvarjen uporabniški gradnik iz prejšnje točke uporabite v glavnem oknu namesto gradnikov, vstavljenih v vaji 1.
3. Za prikaz kode uporabite gradnik RichTextBox namesto gradnika TextBox.

Za izboljšanje uporabniške izkušnje grafičnega uporabniškega vmesnika niste omejeni le na uporabo knjižnice WPF, pogoj je le, da so uporabljene knjižnice zastonj in so na voljo na spletu za prenos. Poskrbite, da se vsi podatki shranjujejo v datoteko.

Vaja 7: Deljenje virov

4t

Pregledovalniku filmske baze poenotite izgled. Vse vizualne lastnosti shranite kot vire:

1. Izdelajte izgled za uporabniški gradnik (glej vajo 6), ki ga shranite v slovar virov (angl. Resource Dictionary). Izgled uporabniškega vmesnika mora biti popolnoma definiran v slovarju virov. Uporabljene barve definirajte kot vire, ter spišite vsaj 3 stile.
2. Poenotite izgled aplikacije z uporabo deljenja virov. Za vsak posamezen gradnik definirajte svoj slovar s stili za specifičen gradnik. Stili naj imajo unikatni ključ!
3. Vmesnik izboljšajte z uporabo drugih uporabniških gradnikov (uporabite tudi druge knjižnice, kot je Extended WPF Toolkit).

Pri oblikovanju grafičnega vmesnika se čim bolj potrudite, izgleda naj lepo, posamezni elementi naj bodo ustrezno postavljeni in barvno skladni z celotnim vmesnikom. Dokončajte vse naloge iz prejšnjih vaj.

Vaja 8: Časovniki in animacije

7t

Aplikaciji dodajte naslednje funkcionalnosti in animacije:

1. V aplikaciji uporabite gradnik GridSplitter, ki omogoča spreminjanje razdelitve prostora med vrsticami ali stolpci.
2. Implementirajte funkcionalnost avtomatskega shranjevanja. V oknu nastavitve omogočite vklop/izklop te funkcionalnosti skupaj s časovno rezino izvajanja. Uporabite časovnik, ki naj sproži shranjevanje vsakih nekaj minut, glede na nastavitve.
3. V aplikacijo umestite tri animacije (uporabite vse tri vrste animacij). Animacije morajo biti smiselne in se skladati z aplikacijo.
4. DODATNO: Po lastni presoji dodajte še kakšno funkcionalnost (časovnik ali animacijo), ki bo smiselno dopolnila celoto in izboljšala uporabniško izkušnjo.

Vaja 9: Usmerjeni dogodki

6t

V tokratni vaji bomo dokončali grafični vmesnik aplikacije:

1. V glavni meni dodajte nov podmeni za spremembo razporeditve. Vsebuje naj dve postavki, prva naj bo privzeta postavitev elementov (kot je ob zagonu aplikacije) in jo označite, druga pa alternativna postavitev. Privzeto je uporabniški gradnik na levi strani ter sam prikaz besedila na desni (levo/desno), medtem ko alternativni pogled spremeni razporeditev gradnikov (spodaj/zgoraj). Ob izbiri v glavnem meniju se posamezni elementi prerazporedijo. Bodite pozorni na prilagajanje velikosti okna (angl. Resize) saj se morajo gradniki prilagajati oknu v obeh pogledih.
2. Poskrbite za brezhibno delovanje in videz vaše aplikacije, da zadovoljite uporabniško izkušnjo.

1. Vaja	XAML	obvezno
2. Vaja	Osnove C#	4t
3. Vaja	Okna	6t
4. Vaja	XML podatki	7t
5. Vaja	Delo s podatki	6t
6. Vaja	Uporabniški gradnik	8t
7. Vaja	Deljenje virov	4t
8. Vaja	Časovniki in animacije	7t
9. Vaja	Usmerjeni dogodki	6t
Ostalo	Design	2t
Skupaj vaje		50t