

IW5 projekt

Důležité

Pro hodnocení projektu (a úspěšné absolvování předmětu) je nutno dokončit **všechny 3 fáze projektu** a projekt **obhájit**. Pokud projekt nebude při obhajobě obsahovat základní funkcionalitu uvedenou v zadání, je možné celkové hodnocení snížit. **Nespokojíme se tedy s nedokončeným projektem**. Tuhle poznámku sem dáváme proto, že se v předchozích ročnících vyskytly týmy, které po dosažení součtu 50 bodů za předmět po 2. fázi rozhodly nedokončit projekt a poté byly nemile překvapeni, když se po nich vyžadovala plná funkcionalita při obhajobě. Dejte si na to tedy prosím pozor.

Zadání

Výsledkem projektu je program na správu hodnocení předmětů IWx (IW1 – IW5).

Seznam předmětů

Jednotlivé předměty (IW1 – IW5) se budou dát procházet jednotlivě.

Seznam studentů

Každý předmět bude obsahovat seznam studentů, kteří jej mají zapsán. Studenti se dají do seznamu přidávat, dá se zobrazit seznam všech studentů a detail jednotlivého studenta. V seznamu studentů je možné vyhledávat podle jména, příjmení a loginu. Seznam studentů je možné seřadit dle jména, příjmení a průběžného hodnocení. V seznamu se dá filtrovat podle dosažené známky (A-F).

Detail studenta

V detailu jsou zobrazeny základní informace, fotka a seznam hodnocení studenta. Informace je možné editovat. Dále pak přidávat nová hodnocení a editovat existující.

Detail hodnocení

Hodnocení se pozůstává z typu (minimálně: cvičení, projekt, zkouška, bonusové body a jiné), bodového hodnocení a slovní poznámky.

Aplikace bude data ukládat do SQL databáze. Data tedy musí zůstat zachována i po ukončení a opětovném spuštění aplikace.

Spolupráce

Projekt řeší studenti v týmech. V každém týmu jsou 3 studenti. Tým o méně studentech není přípustný.

Při řešení projektu týmy využívají GIT na sdílení kódu. Tým do svého GIT repozitáře přidělí přístup vyučujícím (způsob bude vysvětlen v rámci 1. cvičení).

Z GITu musí být viditelná postupná práce na projektu a spolupráce týmu. Pokud uvidíme, že existuje malé množství nelogických a nepřeložitelných commitů, tak nás bude zajímat, jak jste spolupracovali a může to vést na snížení bodového hodnocení.

Do Vašeho týmového projektu si v části Members přidejte účet uciteliw5@outlook.com.

Tento účet budou používat vyučující pro přístup k odevzdávaným souborům. Bez přidání tohoto účtu není možné přistoupit k vašemu projektu, a tedy není možné jej ze strany vyučujících hodnotit.

Návod na přidání člena projektu můžete najít zde: <https://www.visualstudio.com/en-us/docs/setup-admin/team-services/add-team-members-vs>

Odevzdávání

Odevzdávání projektu má 3 fáze. V každé fázi se hodnotí jiné vlastnosti projektu. Nicméně, fáze na sebe navzájem následují a studenti pokračují v práci na svém kódu i po jeho odevzdání v rámci následující fáze.

Kontroluje se kód, který je nahrán v GITu. Vždy se kontroluje poslední commit před časem odevzdávání dané fáze projektu. Na commity nahrány po času odevzdávání nebude brán zřetel.

Je silně doporučováno projekty v průběhu semestru konzultovat s cvičícími, předejdete tak případným komplikacím při odevzdání.

Fáze 1 – objektový návrh (10 bodů) – 19. 3. 2017 23:59:59

V téhle fázi se zaměříme na datový návrh. Vyžaduje se po Vás, aby datový návrh splňoval zadání a nevynechal žádnou podstatnou část. Zamyslete se nad vazbami mezi jednotlivými entitami v datovém modelu. V následující fázi budete entity nahrávat do databáze takže myslíte na jejich propojení již v téhle fázi. V této fázi budeme chtít, abyste odevzdali kód, kde budete mít datové třídy, které budou obsahovat všechny vlastnosti, které budete dále potřebovat a vazby mezi třídami.

Nestačí tedy odevzdat diagram tříd, nebo nějakou jinou reprezentaci. Budeme požadovat kód v jazyce C#.

Pro demonstraci správné funkčnosti tříd vytvořte minimálně 5 unit testů, které budou demonstrovat práci s nimi.

Hodnotíme:

- logický návrh tříd
- využití dědičnosti, zapouzdření, polymorfismu
- nahrávání do GIT po logických částech
- unit testy
- logické rozšíření datového návrhu nad rámec zadání (bonusové body)

Fáze 2 – databáze a WPF backend (20 bodů) – 16. 4. 2017 23:59:59

Aplikace již nepracuje jen s daty uvedenými ve zdrojových souborech. Je napojena na databázi a pracuje s ní. Vytvořte napojení datových tříd pomocí Entity Frameworku na databázi.

V této fázi se od Vás již požaduje vytvoření WPF aplikace. Napište backend aplikace, která bude napojena na Vámi navrhnuté datové modely z 1. fáze a bude schopna načítat a ukládat data do databáze.

Dbejte také kvality Vašeho kódu. Od této fáze se hodnotí i tenhle atribut. Opravte si tedy předchozí kód dle zásad Clean Code a SOLID probíraných na cvičeních a důsledně je dodržujte.

Hodnotíme:

- využití Entity Framework (EF) Code First na vytvoření databáze z tříd navrhnutých ve fázi 1
- návrh WPF aplikace dle návrhového vzoru Model View ViewModel (MVVM)
- čistotu kódu

Fáze 3 – WPF frontend, data binding (40 bodů) – 24 hodin před obhajobou

V poslední fázi vytváříte výslednou podobu Vaší aplikace. Budete zde provazovat backend ve WPF aplikaci, který jste si připravili ve 2. fázi s jednotlivými obrazovkami a zobrazením dat.

Vytvořte View k jednotlivým navrženým ViewModelům. Zamyslete se nad tím, jakým způsobem je vhodné jednotlivá data zobrazovat.

Využijte binding v XAML kódu. Účelem není jenom udělat aplikaci, která funguje, ale také aplikaci, která je správně navržena a může být dále jednoduše upravitelná a rozšiřitelná. Dbejte tedy zásad probíraných ve cvičeních.

Za aplikace, jejichž vizuální návrh bude proveden dobře, a zároveň budou plně funkční, budeme udělovat také bonusové body.

Hodnotíme:

- funkčnost celé výsledné aplikace
- vytvoření View k příslušným ViewModelům z fáze 2
- zobrazení jednotlivých informací dle zadání – seznam studentů, detail...
- správné využití data bindingu v XAML
- čistotu kódu
- vytvoření dobře vypadající a plně funkční aplikace (bonusové body)

Obhajoba

Obhajoby projektů budou probíhat v posledních 2 týdnech výuky cvičení. Termíny obhajob budou vyhlášeny v průběhu semestru.

Na obhajobu se dostaví celý tým. Z členů týmu bude cvičícími vybrán 1 student, který obhajobu povede. Na obhajobu **není nutné** mít prezentaci (powerpoint nebo pdf). Budete nám muset ukázat, jak funguje váš kód, že je správně navržen. Připravte se na naše otázky k funkcionalitě jednotlivých tříd a k důvodům jejich členění. Na obhajobu bude mít tým 15-20 minut.

Finální odevzdání

Po skončení obhajoby bude nutné ještě nahrát váš program do informačního systému. **Bez nahraného programu nemůžeme týmu udělit bodové hodnocení.**