

Závěrečný projekt ASP.NET Core MVC

Cílem projektu je si osvojit a zlepšit znalosti s frameworkm **ASP.NET Core MVC** a případnou jinou volitelnou technologií.

Zároveň Vám poskytuji možnost si vyzkoušet práci v týmu **maximálně o 2 lidech**, která bývá naprostě běžná v praxi. *S větším počtem lidí budou ale samozřejmě růst nároky na kvalitu kódu a množství použitych technologií. Na druhou stranu to přináší značnou výhodu.*

Povinné požadavky pro všechny

- **ASP.NET Core**
 - Verze .NET 8 a výše
- **Entity Framework Core**
 - Database-First přístup (vytvorím databázi v PhpMyAdmin/SQL Management Studio a až potom jdu na kód)
 - Code-First přístup je volitelný (použití migrací - složitější, ale prospěšné do praxe)
- **Relační databáze**
 - MSSQL, MySQL, MariaDB, PostgreSQL

Dodatečné požadavky s vícečlenným týmem

Pokud se rozhodnete pro vícečlenný tým, čeká na Vás více zkušeností, více rozmanité práce a především solidní porce osobního rozvoje. Všechna snaha bude z mé strany i řádně ohodnocena a odměněna. *Za zvolení vícečlenného týmu Vás čeká takové ohodnocení z projektu, které bude sloužit jako hodnocení za celé druhé pololetí.* Zkráceně řečeno, pokud zvládnete projekt, jdete k maturitě :)

Dvoučlenný tým

Povinné požadavky pro dvoučlenný tým

- **Git**
 - Klonování repozitáře, vytváření commitů, větve apod...
 - Jedná se o nahrazení přístupu - pošlu si kód na mail :)
- **Gitové uložiště**
 - GitHub, GitLab, Azure...
- **ASP.NET Core API (pouze Controllers a Model)**
 - ASP.NET Core MVC se použije **pouze jako webové API**, tedy aplikaci, která poskytuje data, nikoliv HTML
- **Frontend technologie**
 - Pro **frontend bude využita jiná technologie**
 - React/Vue/Angular pro webovou aplikaci
 - Případně jejich SSR alternativy jako NextJS, Nuxt
 - Java/Kotlin pro Android mobilní aplikaci
 - Swift/Objective-C pro iOS mobilní aplikaci
 - Avalonia UI pro multiplatform mobilní vývoj
 - Tvorba aplikací pro Android a iOS najednou
 - Na pozadí figuruje hlavně C# a XAML
- **Business logika v C# bude pokryta Unit testy**
 - Bude použit **framework MSTest**
 - Pokud provádíte složitou validační logiku, je dobré mít napsaný test, který kontroluje jeho funkčnost

Nepovinné požadavky pro dvoučlenný tým

- **ASP.NET Core API aplikace může být strukturována do několika menších projektů**
 - ProjectApp.WebApi
 - ProjectApp.Core - entity, business logika, business validace
 - ProjectApp.Persistence - Entity Framework Core
 - ProjectApp.Application - aplikační logika
 - ProjectApp.Infrastructure - odesílání emailů, diskové uložiště apod...

Téma projektu

Témata jsou zcela otevřená vlastním potřebám či fantaziím. Za domácí úkol přes svátky budete mít si vymyslet či vybrat téma, které budete zpracovávat. K tématu se pojí, ale několik pravidel, abych je mohl uznat.

1. **Musí obsahovat autentizaci**
 - Registrace, přihlášení uživatele
2. **Musí obsahovat 2 další databázové entity, které mají mezi sebou vazbu pomocí cizího klíče**
 - Například: filmy + recenze, workouts + jednotlivé cviky, útulky + zvířata...

Dohromady tedy 3 entity. Příkladem může být aplikace, kdy se jedná o evidenci fotbalových týmů a jejich zápasů mezi nimi. Přidávat, upravovat lze pouze až po přihlášení admina.

Jakmile si vyberete téma a případně sestavíte tým, ve kterém to budete řešit, je potřeba téma se mnou prokonzultovat a nechat si jej uznat. V případě, že téma bude moc jednoduché či složité, tak jej upřesníme. **Jakákoliv práce na projektu před schválením tématu nebude uznána.**

Termín pro rozhodnutí na téma projektu a vytvoření týmů je po svátcích první společnou vyučovací hodinu

Příklady témat

- (zakázaný) *ToDoList*- seznam úkolů, evidence splnění, žebříček uživatelů s nejvíce splněnými úkoly
- *Workout App* - zaznamenávání tréninků, jejich trvání, jaké cviky se prováděli, žebříček lidí s nejvíce tréninky za uběhlý měsíc
- *Daily Dopamine Booster* - odklikávací aplikace jednou denně, uživatelé dostanou XP za odkliknutí, žebříček TOP hráčů
- *Fakturovač* - vytváření faktur, evidence položek na faktuře, tisk do PDF