

# ASP.NET Core

## Индивидуално задание

### Основни изисквания

Приложението трябва да използва следните технологии, фреймуърци и техники за програмиране:

- Приложението трябва да имплементира **ASP.NET Core Framework**.
  - Приложението трябва да има поне **10 изгледа (views)**
  - Приложението трябва да има поне **5 самостоятелни** ентити модела, да бъдат използвани релации между таблиците (**one-to-many, one-to-one, many-to-many**)
  - Приложението трябва да има поне **5 контролера**
  - Да има пълни **CRUD** функционалности за поне **3 ентитета**
- Използвайте **Visual Studio 2019 / Visual Studio 2022 / JetBrains Project Rider**.
- Използвайте **Microsoft SQL Server** като база данни
- Използвайте **Entity Framework Core** за достъп до базата
- Използвайте стандартната **ASP.NET Identity System** за управление на потребители и роли
  - Поне един от потребителите трябва да има роли: **NormalUser** и **Administrator**
- Внедрете **error handling** и **data validation** за да избегнете крашванията, когато са въведени невалидни данни от клиентска страна

Проектът трябва да е с добре структурирана архитектура и контрол над кода

- Препоръчително е придържането към най-добрите практики за ООП дизайн и висококачествен код:
  - SOLID принципи (data encapsulation, inheritance, abstraction, etc.)
  - Strong cohesion и loose coupling
  - Придържане към конвенциите за именуване в езика C#
  - Кодът трябва да е добре подреден и четим



## Системи за управление на версиите

Използвайте система за управление на версиите: **GitHub, BitBucket, Gitlab**

- Линк към вашият проект
- Трябва да имате **commits** в поне **3 различни дни**
- Трябва да имате поне **10 commits**

**Внимание:** Изискванията за система за управление е абсолютно задължителна.

**Внимание:** Неспазването на изискванията за системата на управление на версиите ще понижи вашата оценка.

## Защита

Всеки студент ще има защита на своя проект пред изпитващия.

Студентът ще има **само 10-15 мин** за :

- **Демонстрация** как работи проектът
- Показване на кода и кратко обяснение
- Отговор на въпроси, свързани с проекта и тематиката на курса

## Бонуси

- Всичко, което не е свързано с по-горните изисквания, но е интегрирано като допълнение в проекта
- Примери
  - **Сийдване на базата** с примерни данни
  - Използване на **Repository-Service** патърна
  - Имплементация на функционалности като **pagination, сортиране на данни, търсене и филтър по различни критерии**
  - Отделяне на логика в **MVC Areas** и **partial view-та**.

