**Спецификација**

за изработка на проектна задача

**Вовед**

Во проектната задача потребно е да се изработи ASP.NET Core MVC апликација за управување со предмети, студенти и наставници на универзитет/факултет/училиште. Податочниот дел од апликацијата се состои од следните ентитети: студенти, наставници, предмети, како и врзувачки ентитети за студенти и предмети. Слика 1 го прикажува изгледот на базата на податоци, потребните табели и својства, како и врските помеѓу ентитетите со надворешни клучеви. Врз база на прикажаната структура на базата, ќе треба да развиете EF Core модели за управување со податоците и да креирате база во вашиот MVC проект. Дополнително ќе треба да развиете соодветни контролери и погледи за управување со податоците. Во следните фази (наредните недели) иницијалната MVC апликација ќе треба да биде проширена со WEB API за управување со податоците, како и со нови функционалности кои ќе ви бидат дополнително специфицирани.

**Начин и динамика на изработка на проектната задача**

Проектот ќе се работи индивидуално. Секој треба да креира и развива своја верзија. Проектот е задолжителен за сите студенти и ќе се работи во три фази (milestones) во текот на неколку недели. Секоја фаза ќе трае две недели. Првата фаза ќе биде елаборирана во наредното поглавје, а следните фази и патокази ќе бидат објавувани дополнително во наредните недели.

**Фаза 1**

Во првата фаза од проектната задача, потребно е врз основа на структурата на податочната база прикажана на Слика 1, да се развие иницијалната верзија на MVC апликацијата (со code first пристап), која ќе овозможи управување со ентитетите. Поконкретно, првата фаза треба да ги покрие следните активности/развојни чекори:

1. Развој на податочни модели врз база на спецификацијата на базата. Поврзување на ентитетите со соодветните one-to-many и many-to-many врски. Треба да се овозможат следните поврзувања:
   1. Еден предмет може да биде врзан со двајца наставници, преку надворешните клучеви FirstTeacherID и SecondTeacherID. Еден наставник може да биде врзан за повеќе предмети, како прв или втор наставник.
   2. Еден студент може да е запишан на повеќе предмети, а на еден предмет може да има запишано повеќе студенти. Оваа many-to-many релација е овозможена со junction табелата Enrollment, каде CourseId и StudentId се надворешните клучеви кон предмет и студент, соодветно.
2. Развој на соодветните MVC контролери и погледи за моделите. Контролерите треба да подржат акции за вметнување на нови ентитети, модификација и бришење на постоечки ентитети, листање на ентитетите, како и поврзување на соодветните ентитети. Погледите претставуваат кориснички интерфејс за спроведување на горенаведените акции. Треба да бидат овозможени следните акции и погледи во MVC апликацијата:
   1. Преглед на сите предмети, со можности за филтрирање по наслов, семестар и програма;
   2. Преглед на предмети *по наставник*;
   3. Погледи/акции за додавање на нов предмет, модификација, бришење и детали на постоечки предмети;
   4. Преглед на наставници, со можности за филтрирање по име и презиме, степен на образование и академски ранг;
   5. Погледи/акции за додавање на нов наставник, модификација, бришење и детали за постоечки наставници;
   6. Преглед на сите студенти, со можности за филтрирање по индекс и име и презиме;
   7. Преглед на студенти *по предмет*;
   8. Погледи/акции за додавање на нов студент, модификација, бришење и детали за постоечки студенти.

Во однос на поврзаните податоци:

* 1. Врската помеѓу предмет и наставници може да се управува во погледите/акциите за додавање на нов предмет и модификација на постоечки предмет.
  2. Врската помеѓу предмет и студенти може да се управува во погледите/акциите за модификација на предмет. Да се разгледа и можноста да се дефинираат нови и засебни акции/погледи за запишување (enrollment) на студенти на даден предмет. Да се разгледа и можност за модификација на врската од аспект на внесување и едитирање на останатите полиња (освен надворешните клучеви) во junction табелата Enrollment. Посебен акцент да се даде на оваа точка. Студентите може да понудат и свои различни решенија кои се компатибилни со ова барање. Најиновативното решение ќе биде наградено.

Откога ќе ја завршите првата фаза потребно е да ни го испратите вашиот проект (или линк од github) на [danield@feit.ukim.edu.mk](mailto:danield@feit.ukim.edu.mk) и на [pero@feit.ukim.edu.mk](mailto:pero@feit.ukim.edu.mk). Рок за изработка на оваа фаза е две недели, т.е. краен рок за испраќање е 23.04.2020.



Слика 1: База на податоци за .NET Core MVC апликацијата со потребните табели, спецификации за колоните и врски помеѓу табелите