ASP.NET Core – December 2020

Individual Project Assignment

19 December 2020

**1. General Requirements**

Your Web application should use the following technologies, frameworks and development techniques:

 The application must be implemented using ASP.NET Core Framework (latest).

o The application must have at least 10 web pages (views)

o The application must have at least 5 entity models

o The application must have at least 5 controllers

 Use Visual Studio 2019 / JetBrains Project Rider.

o Use the Razor template engine for generating the UI

 Use sections and partial views.

 Use display and editor templates.

o Optionally, you could also use Web API to create a RESTful service and use JavaScript / TypeScript

for the Front-End

 Use Microsoft SQL Server as Database Service

o Optionally, use multiple storages, e.g. files, other Web services, databases (e.g. MySQL / MongoDB /

Cassandra / etc.)

 Use Entity Framework Core to access your database

o If you need additional connectors to other databases, feel free to use them

 Use MVC Areas to separate different parts of your application (e.g. area for administration)

 Adapt the default ASP.NET Core site template or get another free theme

o Use responsive design based on Twitter Bootstrap / Google Material design

o Or just design your own

 Use the standard ASP.NET Identity System for managing Users and Roles

o Your registered users should have at least one of these roles: User and Administrator

o If you need, implement your own user management system

 Optionally, use AJAX request to asynchronously load and display data somewhere in your application

 Write Unit Tests for your logic, controllers, actions, helpers, etc.

o You should cover at least 80% of your business logic.

 Implement error handling and data validation to avoid crashes when invalid data is entered

o Both client-side and server-side, even at the database(s)

 Handle correctly the special HTML characters and tags like &lt;br /&gt; and &lt;script&gt; (escape special characters)

 Use Dependency Injection

o The built-in one in ASP.NET Core is perfectly fine

 Optionally, use AutoМapping

 Prevent from security vulnerabilities like SQL Injection, XSS, CSRF, parameter tampering, etc.

 DO NOT use the project developed during the lectures by the lecturer. Try to do something different.

**2. Additional Requirements**

Your Project MUST have a well-structured Architecture and a well-configured Control Flow.

© SoftUni – about.softuni.bg. Copyrighted document. Unauthorized copy, reproduction or use is not permitted.

Follow us: Page 3 of 3

 Follow the best practices for Object Oriented design and high-quality code for the Web application:

o Use the OOP principles properly: data encapsulation, inheritance, abstraction and polymorphism

o Use exception handling properly

o Follow the principles of strong cohesion and loose coupling

o Correctly format and structure your code, name your identifiers and make the code readable

 Make the user interface (UI) good-looking and easy to use

o If you provide a broken design, your Functionality Points will be sanctioned

 Support all major modern Web browsers

o Optionally, make the site as responsive as possible – think about tablets and smartphones

 Use Caching where appropriate

**3. Source Control**

Use a source control system by choice, e.g. GitHub, BitBucket

 Submit a link to your public source code repository

 You should have commits in at least 5 DIFFERENT days

 You should have at least 20 commits

IMPORTANT: The Source Control Requirements are ABSOLUTELY MANDATORY.

IMPORTANT: NOT following the Source Control Requirements will result in your DIRECT DISQUALIFICATION from

the Project Defenses.

**4. Public Project Defense**

Each student will have to deliver a public defense of its work in front of a trainer.

Students will have only 10-15 minutes for the following:

 Demonstrate how the application works (very shortly)

 Show the source code and explain how it works

 Answer questions related to the project (and best practices in general)

Please be strict in timing! On the 15 th minute you will be interrupted! It is good idea to leave the last 2-3 minutes

for questions from the trainers.

Be well prepared for presenting maximum of your work for minimum time. Bring your OWN LAPTOP. Test it

preliminarily with the multimedia projector. Open the project assets beforehand to save time.

**5. Bonuses**

 Anything that is not described in the assignment is a bonus if it has some practical use

 Examples

o Use SignalR communication somewhere in your application.

o Use Front-End Frameworks (like Angular, React, Blazor)

o Host the application in a cloud environment, e.g. in AppHarbor or Azure

o Use a file storage cloud API, e.g. Dropbox, Google Drive or other for storing the files

o Use of features of HTML5 like Geolocation, Local Storage, SVG, Canvas, etc.

**6. Assessment Criteria**

 Functionality – 0…30

© SoftUni – about.softuni.bg. Copyrighted document. Unauthorized copy, reproduction or use is not permitted.

Follow us: Page 3 of 3

 Implementing controllers correctly (controllers should do only their work) – 0...5

 Implementing views correctly (using display and editor templates) – 0…10

 Unit tests (unit test for some of the controllers using mocking) – 0…10

 Security (prevent SQL injection, XSS, CSRF, parameter tampering, etc.) – 0…5

 Data validation (validation in the models and input models) – 0…10

 Code quality (well-structured code, following the MVC pattern, following SOLID principles, etc.) – 0…10

 Bonus (bonus points are given for exceptional project) – 0…20