

CURSO	Licenciatura em Informática		
UNIDADE CURRICULAR	Tecnologias de Internet III	Obrigatória	X
		Opcional	
ÁREA CIENTÍFICA	Engenharia Informática		

Ano: 2º	Semestre: 1º	ECTS: 4	Horas de Contacto teórico práticas: 60h
---------	--------------	---------	---

OBJETIVOS DA APRENDIZAGEM

Para concluir com sucesso esta unidade curricular, os alunos deverão demonstrar possuir os seguintes conhecimentos e capacidades:

1. Compreender a arquitetura cliente-servidor das aplicações web;
2. Conhecer a tecnologia .NET para a criação de aplicações web;
3. Instalar e configurar o ambiente de desenvolvimento web para ASP.NET Core;
4. Compreender o paradigma MVC-Model View Controller para o desenvolvimento de aplicações web;
5. Entender a função dos controllers na tecnologia ASP.NET Core e saber utilizá-los no desenvolvimento de aplicações;
6. Compreender a função das views e saber utilizá-las no desenvolvimento de aplicações ASP.NET Core;
7. Compreender a função dos models e saber utilizá-los na criação de aplicações web;
8. Saber utilizar as funções de middleware routing no desenvolvimento de aplicações;
9. Saber criar aplicações de acesso a bases de dados com ADO.NET e de acordo com o paradigma MVC;
10. Saber criar aplicações com base no modelo razor pages;
11. Saber utilizar a bootstrap em aplicações ASP.NET;
12. Saber criar aplicações para a Azure e para Linux.

PROGRAMA

1. ASP.NET Core – Visão geral
 - 1.1. As aplicações web
 - 1.2. Client-side e Server-side
 - 1.3. Métodos HTTP
 - 1.4. A ASP.NET Core MVC
2. Configurar o ambiente de desenvolvimento
 - 2.1. Instalação do Visual Studio
 - 2.2. Instalação do Visual Studio Core
 - 2.3. Instalação do .NET Core SDK
3. O paradigma MVC-Model View Controller
 - 3.1. Criar um novo projeto ASP.NET Core MVC
 - 3.2. O padrão MVC – Models, Controllers e Views
 - 3.3. Os projetos ASP.NET Core – Ficheiros e pastas do projeto
4. Controllers
 - 4.1. Papel dos Controllers nas aplicações ASP.NET Core MVC
 - 4.2. Criar um primeiro Controller
 - 4.3. Adicionar Views
 - 4.4. Adicionar Models
 - 4.5. Passar dados do Controller para a View
5. Views
 - 5.1. A *View engine* e a *Razor View engine*
 - 5.2. Layouts
 - 5.3. Gerar HTML
 - 5.4. Componentes de uma View
 - 5.5. Tag Helpers
6. Models
 - 6.1. Models
 - 6.2. Ligação Controller-Model
 - 6.3. Utilizar Entity Framework em aplicações ASP.NET MVC
7. Validação de dados
 - 7.1. Visão geral
 - 7.2. Validação *Server-side*
 - 7.3. Validação *Client-side*
8. Routing
 - 8.1. Convention-based routing
 - 8.2. Attribute-based routing
 - 8.3. Atributo Route ao nível do *Controller*
 - 8.4. Passar valores do routing no HTTP
 - 8.5. Restrições
9. ADO.NET – Acesso a fontes de dados
 - 9.1. Visão geral
 - 9.2. Tipos de dados
 - 9.3. DataSets, DataTables e DataViews
 - 9.4. Entity Framework
10. Criar uma MVC web app
 - 10.1. Adicionar um *Controller*

- 10.2. Adicionar uma *View*
- 10.3. Adicionar um *Model*
- 10.4. Trabalhar com a *SQL Server LocalDB*
- 11. Criar aplicações com base no modelo Razor Pages
 - 11.1. Uma introdução às *Razor Pages*
 - 11.2. Adicionar um *Model*
- 12. Utilizar a Bootstrap em aplicações ASP.NET Core
 - 12.1. O código HTML e CSS
 - 12.2. A biblioteca Bootstrap
 - 12.3. Utilizar a Bootstrap em aplicações ASP.NET Core
- 13. *Deployment* de aplicações ASP.NET Core
 - 13.1. O ficheiro *project.json*
 - 13.2. Deployment para Azure
 - 13.3. Deployment para ambiente Linux

DEMONSTRAÇÃO DE COERÊNCIA ENTRE CONTEÚDOS PROGRAMÁTICOS E RESULTADOS DA APRENDIZAGEM

Os conteúdos 1.1 a 1.4 e 2.1 a 2.3 permitem atingir os objetivos 1, 2 e 3. O objetivo 4 é concretizado através dos pontos 3.1 a 3.3. Os pontos 4.1 a 4.3 correspondem ao objetivo 5. O objetivo 6 é concretizado através dos pontos 5.1 a 5.5. Os pontos 6.1 a 6.3 permitem alcançar o objetivo 7. O objetivo 8 é concretizado através dos pontos 8.1 a 8.5. O objetivo 9 é concretizado através dos pontos 9.1 a 9.4 e 10.1 a 10.4. Os pontos 11.1 E 11.2 permitem concretizar o objetivo 10. O objetivo 11 é atingido pelo ponto 12. O ponto 13 corresponde ao objetivo 12.

METODOLOGIA DE ENSINO E AVALIAÇÃO

As aulas são de natureza teórico-prática. Estão previstas 60 horas de contato. O tempo total de trabalho do aluno corresponde a 108 horas. O desenvolvimento das matérias é acompanhado da realização de exercícios e pequenos projetos que ilustram os conceitos e as tecnologias apresentadas. Desta forma, os alunos adquirem a compreensão teórica e prática que lhes possibilita, como objetivo final, desenvolver aplicações web com a tecnologia ASP.NET.

De acordo com o Regulamento de Funcionamento do ISTECS a avaliação é efetuada através de um exame escrito individual e obrigatório. Na classificação final, poderão ser considerados elementos de avaliação contínua, tais como testes, trabalhos individuais ou em grupo, assim como a participação nas aulas presenciais e em recursos de aprendizagem proporcionados por sistemas de e-learning.

DEMONSTRAÇÃO DE COERÊNCIA ENTRE METODOLOGIAS DE ENSINO E RESULTADOS DE APRENDIZAGEM

A metodologia usada nesta unidade curricular, baseada na criação de aplicações web de complexidade progressiva, é implementada para que os alunos adquiram as competências que

permitam atingir o objetivo fundamental que consiste na capacidade para desenvolver aplicações Web usando as tecnologias ASP.NET e ADO.NET.

BIBLIOGRAFIA

Fundamental:

Ragupathi, Mugilan T. S. Learning ASP.NET Core MVC Programming, Packt Publishing
Ragupathi, Mugilan T. S.; De Sanctis, Valerio; Singleton, James, ASP.NET Core: Cloud-ready, Enterprise Web Application Development, Packt Publishing
Ciliberti, John. ASP.NET Core Recipes: A Problem-Solution Approach, Apress

Complementar:

Luis Abreu; ASP.NET 4.5 - CURSO COMPLETO; Editora FCA.
Luís Abreu. ASP.NET 4.5.1 - ONE ASP.NET, OWIN, IDENTITY, MVC, WEB API E SIGNALR . FCA
Ken Cox . ASP.NET 3.5 Para Leigos. Alta Books;
Tim Patrick. Microsoft ADO.NET 4 Step by Step (Step By Step (Microsoft)).
Imar Spaanjaars . Beginning ASP.NET 4.5.1: in C# and VB . (Wrox Programmer to Programmer) ;
Andrew Duthie, Matthew MacDonald. ASP.NET in a Nutshell. ;
Freeman, Adam. Pro ASP.NET Core MVC 2, Apress

INTERNET:

<https://docs.microsoft.com/en-us/aspnet/core/>
Acesso a publicações da especialidade, gratuitamente, através da rede SPRINGER:
<https://link.springer.com/>