

PROGRAMAÇÃO WEB AVANÇADA

Mestrado em Engenharia Informática

Código: 26205

Área Científica Predominante: Ciências da Computação

Docente: Paulo Rafael da Costa e Sousa

Idioma de Instrução: Português

Regime: S2

Carga Letiva: 60h Carga Trabalho: 100h

ECTS: 6,0

Objetivos

O desenvolvimento de aplicações para a Web e uma atividade que combina, em igual proporção, o design, a usabilidade e a tecnologia. Para ser proficiente no desenvolvimento de aplicações Web, é fundamental dominar aspetos tão variados como protocolos de comunicação, engenharia de software, gestão de redes, design de interfaces, bases de dados, linguagens de anotação, scripting, segurança, legislação, entre outros.

Esta unidade curricular visa fornecer aos alunos os conhecimentos avançados para o desenvolvimento de aplicações Web modernas, que sejam simultaneamente seguras e eficazes (do ponto de vista da usabilidade e desempenho), tanto numa perspetiva de Front-end e Back-end, como de DevOps.

Os resultados serão alcançados gradualmente ao longo do semestre, através de exercícios práticos desenvolvidos em grupo, onde os estudantes são encorajados a pesquisar e desenvolver estratégias para implementar as tecnologias estudadas nas aulas.

Resultados da Aprendizagem

No final da unidade curricular os alunos deverão ser capazes de:

Compreender o protocolo HTTP

Compreender as diferenças entre os vários métodos e códigos de erro HTTP

Conhecer as linguagens fundamentais de programação para a Web

Construir um sítio Web a partir de um mockup de design

Conhecer e utilizar eficazmente frameworks CSS e Javascript

Desenvolver aplicações server-side, nomeadamente utilizando a framework NestJS

Conhecer Web services de acesso público e integrar os mesmos nas suas aplicações

Desenvolver Web services segundo a metodologia RESTful

Desenvolver aplicações Web cliente-side recorrendo a frameworks de Javascript

Reconhecer quando e como utilizar bases de dados NoSQL

Conhecer estratégias de deployment automático

Conhecer serviços Cloud para deployment de aplicações Web

Conteúdos Programáticos

01 – Introdução a Web e ao protocolo HTTP

Standards W3C

Protocolo HTTP

Componentes da Web

Pedidos HTTP

Status codes

Formato de mensagens/MIME types

02 – Javascript e introdução ao NodeJS

Tipos de dados

Operadores: atribuição, aritméticos, binários, lógicos, precedência, etc.

Expressões, estruturas de decisão, estruturas de repetição

Estruturas de dados (Objetos e Arrays)

Funções nativas

ECMA2015/ES6

Classes e funções

Browser objects

Introdução ao NodeJS

Ecossistema do NodeJS

Node Package Manager (npm)

Anatomia de uma aplicação NodeJS

03 – Tópicos Avançados de Javascript

Modulos

Callbacks vs Promises vs Async

04 - Programação server-side: NestJS

O que é?

Estrutura em layers: Controllers, Services, Models

Providers

Modules

Scaffolding de aplicações NestJS

Geração de resources CRUD

Controllers

05 – Arquiteturas orientadas ao serviço

Introdução as Arquiteturas Orientadas ao Serviço (SOA)

SOAP vs REST

RESTful vs RESTless

Operações CRUD

Boas práticas de desenvolvimento de serviços RESTful

Tratamento de erros

06 - Aula de acompanhamento e apoio ao desenvolvimento do Trabalho Prático

Espaço para reflexão e discussão e desenvolvimento de soluções no âmbito da UC

Modelo de domínio

07 – Sistemas de indexação escaláveis e bases de dados NoSQL

Introdução ao paradigma NoSQL

Análise comparativa de bases de dados NoSQL

Bases de dados orientadas ao documento (MongoDB)

Operações CRUD em MongoDB

Principais operadores

Tipos de índices

Agregações

Comunicação NodeJS/Mongo (Mongoose)

08 – Showcase da plataforma Supabase

Videoconferência com engenheiros da Supabase

Apresentação da plataforma

Explicação de aplicação de demonstração

Apresentação de uma solução (código) e resultado final

QA

09 – Programação client-side: XHTML, HTML5, CSS 3 e React

Anatomia de um documento HTML

XHTML

Elementos introduzidos pelo HTML 5

Seletores CSS

Prioridade das regras CSS

Box model

Formatação de texto

Posicionamento de elementos

Seletores avançados

Pseudo classes

CSS 3

Design responsivo

Frameworks de CSS

Introdução ao React

Scaffolding de uma aplicação React

Anatomia de uma aplicação React

JSX

Componentes React

Estado do componente

Passagem de parâmetros

Tipagem de parametros

Eventos

Depuracao de aplicacoes React

10 – Programacao client-side (continuacao)

Ciclo de vida de um componente React

Tratamento de formularios

Roteamento

Hooks

11 - Aula de acompanhamento e apoio ao desenvolvimento do Trabalho Pratico

Espaco para reflexao e discussao e desenvolvimento de solucoes no ambito da UC

Push notifications

12 – DevOps

Introducao ao conceito de DevOps

Ferramentas de apoio a atividade de DevOps

Construcao de containers Docker

Composicao de containers Docker

Deployment de containers Docker

13 - Aula de acompanhamento e apoio ao desenvolvimento do Trabalho Pratico

Espaco para reflexao e discussao e desenvolvimento de solucoes no ambito da UC

Interceptors axios

14 - Aula de acompanhamento e apoio ao desenvolvimento do Trabalho Pratico

Espaco para reflexao e discussao e desenvolvimento de solucoes no ambito da UC

15 – Exame

Exame final

Bibliografia Recomendada

ReactJS by Example – Building Modern Web Applications with React. Vipul A. M., Prathamesh Sonpatki. Packt

Publishing, 2016.

Full Stack Javascript, Azat Mardan, APRESS

Enterprise Web Development - Building HTML5 Applications: From Desktop to Mobile

Gourley, D., Totty, B. (2002). HTTP: The definitive guide; [understanding web internals]. Beijing, u.a.: O'Reilly.

Crockford, D. (2008). JavaScript: The good parts. Beijing: O'Reilly.

Métodos de Ensino e de Aprendizagem

As metodologias de ensino e de aprendizagem empregues nesta UC promovem a assimilação dos conhecimentos lecionados através da realização de exercícios práticos e estimulam as capacidades de investigação e de auto-aprendizagem dos alunos através da aplicação de desafios para resolução individual e/ou em grupo apoiada pelo docente.

Métodos de Avaliação

Avaliação em época normal

A avaliação desta unidade curricular é constituída por 2 momentos de avaliação: 1 trabalho prático realizado em grupos e fora do horário letivo, e um momento de avaliação individual que assumirá a forma de um exame com nota mínima de 8 valores:

TP – Trabalho de grupo

E – Exame individual (nota mínima: 8 valores em 20)

O trabalho prático será avaliado segundo diversos critérios definidos no enunciado do próprio trabalho, no entanto, pelo menos as seguintes componentes farão parte dos critérios de avaliação:

completude e qualidade da solução apresentada,

qualidade do relatório escrito, e

qualidade da apresentação/defesa do trabalho (avaliação individual).

A nota final atribuída a cada aluno corresponde à seguinte fórmula:

$$\text{Nota final} = \text{TP} * 0.4 + \text{E} * 0.6$$

A deteção de fraude em algum dos trabalhos implica a reprovação à unidade curricular.

Época de recurso

A avaliação E pode ser repetida na época de recurso, onde se mantêm as mesmas regras e fórmula de cálculo da nota final.

Nota: a avaliação TP é feita, exclusivamente, durante a época normal.