
Ano Letivo 2022-23

Unidade Curricular DESENVOLVIMENTO DE APLICAÇÕES PARA A WEB

Cursos ENGENHARIA INFORMÁTICA (1.º ciclo)

Unidade Orgânica Faculdade de Ciências e Tecnologia

Código da Unidade Curricular 14781058

Área Científica CIÊNCIA DE COMPUTADORES

Sigla

Código CNAEF (3 dígitos) 481

Contributo para os Objetivos de Desenvolvimento Sustentável - ODS (Indicar até 3 objetivos) 4,8,10

Línguas de Aprendizagem Português

Modalidade de ensino

Presencial

Docente Responsável

Noélia Susana Costa Correia

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Noélia Susana Costa Correia	PL; T	T1; PL1; PL2	28T; 56PL

* Para turmas lecionadas conjuntamente, apenas é contabilizada a carga horária de uma delas.

ANO	PERÍODO DE FUNCIONAMENTO*	HORAS DE CONTACTO	HORAS TOTAIS DE TRABALHO	ECTS
3º	S1	28T; 28PL	156	6

* A-Anual;S-Semestral;Q-Quadrimestral;T-Trimestral

Precedências

Sem precedências

Conhecimentos Prévios recomendados

Bases de Dados, Programação Imperativa, Redes de Computadores

Objetivos de aprendizagem (conhecimentos, aptidões e competências)

No final da disciplina o aluno deverá dominar conceitos e técnicas utilizadas para o desenvolvimento de aplicações baseadas na web:

- implementar o backend de uma aplicação web, em node.js, suportada por uma base de dados;
- implementar o frontend de uma aplicação web, em react;
- conhecimento de frameworks uteis em várias etapas do desenvolvimento;
- adquirir noções de otimização e autenticação/validação/segurança aquando da interação cliente-servidor.

Conteúdos programáticos

1. Introdução

- TypeScript (TS)
- Organização de um projeto TS
- Ambiente de desenvolvimento: sistemas de modulos e utilização de agregadores (bundlers) para maior eficiencia.

2. Lado servidor

- Node.js e NPM
- Built-in and non-built in libraries/modules/frameworks
- REST APIs
- DB access

3. Lado cliente

- React
- JSX
- Babel
- Webpack

4. Containers e Containerização

Metodologias de ensino (avaliação incluída)

A leccionação das aulas teóricas têm como suporte um conjunto de slides. Os conteúdos serão sedimentados pela realização de pequenos programas e demonstração/implementação de um projeto alargado que integra os vários conhecimentos. Os alunos farão posteriormente o seu projeto, o qual estará sujeito a avaliação e discussão.

Avaliação: A pontuação final é a média ponderada do exame escrito (50%) e projeto (50%). Os alunos estão admitidos a exame se entregarem o projeto prático até ao último dia de aulas. Ainda em relação ao projeto:

- É aceite se cumprir os objetivos.
- Terá de ser realizado individualmente, inclui um relatório e terá de ser discutido.

Os critérios de admissão são aplicados a todos os alunos, incluindo os trabalhadores-estudantes e dirigentes associativos.

Bibliografia principal

Frank Zammetti, ?Modern Full-Stack Development?, 2020.

Vários recursos disponíveis na web, cujos links estarão nos slides (ex: typescript-handbook.pdf)

Academic Year 2022-23

Course unit DEVELOPMENT OF APPLICATIONS FOR THE WEB

Courses INFORMATICS (COMPUTER SCIENCE) (1st Cycle)

Faculty / School FACULTY OF SCIENCES AND TECHNOLOGY

Main Scientific Area

Acronym

CNAEF code (3 digits) 481

Contribution to Sustainable Development Goals - SGD (Designate up to 3 objectives) 4,8,10

Language of instruction Portuguese

Teaching/Learning modality Presencial

Coordinating teacher Noélia Susana Costa Correia

Teaching staff	Type	Classes	Hours (*)
Noélia Susana Costa Correia	PL; T	T1; PL1; PL2	28T; 56PL

* For classes taught jointly, it is only accounted the workload of one.

Contact hours	T	TP	PL	TC	S	E	OT	O	Total
	28	0	28	0	0	0	0	0	156

T - Theoretical; TP - Theoretical and practical ; PL - Practical and laboratorial; TC - Field Work; S - Seminar; E - Training; OT - Tutorial; O - Other

Pre-requisites

no pre-requisites

Prior knowledge and skills

Database Design, basic programming skills, computer networks and protocols.

The students intended learning outcomes (knowledge, skills and competences)

At the end of the course the student should master concepts and techniques used for the development of web-based applications:

- implement the backend of a web application, in node.js, supported by a database;
- implement the frontend of a web application, in react;
- knowledge of frameworks useful in various stages of development;
- acquire notions of optimization and authentication/validation/security during client-server interaction.

Syllabus

1. introduction

- TypeScript (TS)
- Organization of a TS project
- Development environment: module systems and use of bundlers for greater efficiency.

2. Server side

- Node.js and NPM
- Built-in and non-built in libraries/modules/frameworks
- REST APIs
- DB access

3. Client side

- React
- JSX
- Babel
- Webpack

4. Containers and Containerization

Teaching methodologies (including evaluation)

The lecture classes will be supported by a set of slides. The contents will be sedimented by the realization of small programs and demonstration/implementation of a project that integrates the various knowledge. Students will then make their project, which will be subject to assessment and discussion.

Assessment: The final mark is the weighted average of the written exam (50%) and project (50%). Students are admitted to the exam if they deliver the practical project by the last day of classes. Still in relation to the project:

- It is accepted if it meets the objectives.
- It has to be done individually, includes a report and has to be discussed.

The admission criteria are applied to all students, including student workers and association leaders.

Main Bibliography

Frank Zammetti, ?Modern Full-Stack Development?, 2020.

Vários recursos disponíveis na web, cujos links estarão nos slides (ex: typescript-handbook.pdf)
