

## Ficha de Unidade Curricular – (Versão A3ES 2018-2023)

### 1. Caracterização da Unidade Curricular.

- 1.1. **Designação da unidade curricular** (1.000 carateres).  
Introdução à Programação na Web / Introduction to Internet Programming
- 1.2. **Sigla da área científica em que se insere** (100 carateres).  
IC
- 1.3. **Duração<sup>1</sup>** (100 carateres).  
Semestral
- 1.4. **Horas de trabalho<sup>2</sup>** (100 carateres).  
162 h
- 1.5. **Horas de contacto<sup>3</sup>** (100 carateres).  
TP 67,5 h
- 1.6. **ECTS** (100 carateres).  
6
- 1.7. **Observações<sup>4</sup>** (1.000 carateres).  
Partilhada com outros cursos.
- 1.7. **Remarks** (1.000 carateres).  
Common with other courses.

2. **Docente responsável e respetiva carga letiva na Unidade Curricular** (preencher o nome completo) (1.000 carateres).  
José Luís Falcão Cascalheira

3. **Outros docentes e respetivas cargas letivas na unidade curricular** (1.000 carateres).

4. **Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes).** (1.000 carateres).

Os estudantes que terminam com sucesso esta unidade curricular serão capazes de:

1. Enquadrar os principais elementos constituintes da arquitetura da World Wide Web;
2. Demonstrar conhecimento teórico-prático sobre as principais normas associadas à componente de cliente (web browser);
3. Utilizar o modelo de programação na componente cliente, nomeadamente no acesso dinâmico a conteúdos e tratamento de eventos;
4. Utilizar e estender uma tecnologia de servidor **com significativa adoção industrial**, para criação de aplicações Web de pequena complexidade;
5. Desenhar e implementar aplicações web com manutenção de estado de conversação, visualização e edição de dados.

4. **Intended learning outcomes (knowledge, skills and competences to be developed by the students).** (1.000 characters).

Students who successfully complete this course unit be able to:

1. Frame the main constituent elements of the World Wide Web architecture
2. Demonstrate theoretical and practical knowledge on key standards associated with the client component (web browser);
3. Use the programming model in the client component, namely in dynamic access to content and event handling;
4. Use and extend server technology **with significant industrial adoption**, to create low complexity Web applications;
5. Design and implement web applications with maintenance of conversation status, visualization and data editing.

**5. Conteúdos programáticos (1.000 carateres).**

- I. A arquitetura da *World Wide Web*: Identificação de recursos (URIs), interação (protocolo HTTP) e representação.
- II. Distribuição de conteúdos web (protocolo HTTP).
- III. Infraestruturas de suporte à criação da interface com o utilizador em aplicações web e respetivo modelo de programação.
- IV. Componente de cliente (Web browser): descrição da estrutura (HTML), formatação visual (CSS), manipulação programática e atualizações totais e parciais da interface gráfica.
- V. Componente de servidor: distribuição de conteúdos estáticos; geração dinâmica de conteúdos; modelo de programação no servidor, utilizando o padrão MVC; manutenção de estado (de visualização, de sessão e de aplicação); interceção de pedidos.
- VI. Consequências da distribuição no modelo de programação e respetivas arquiteturas de referência.
- VII. Concretização dos temas estudados através do desenvolvimento de aplicações Web de pequena e média complexidade com suporte para persistência de dados.

**5. Syllabus (1.000 characters).**

- I. Architecture of the World Wide Web: Resource Identification (URI), interaction (HTTP) and representation (HTML).
- II. Distribution of web content (HTTP protocol).
- III. Support Infrastructures for the creation of user interface in web applications and their programming model.
- IV. Client Component (web browser): description, visual formatting, programmatic manipulation and total and partial updates of the GUI.
- V. Server component: static content distribution, dynamic generation of content; programming model on the server, using the MVC pattern; maintaining state (view, session and application); HTTP requests intercepting.
- VI. Distribution consequences in the programming model and corresponding reference architectures. Caching mechanisms.
- VII. Apply the studied subjects by developing small and medium complexity web applications with support for data persistence.

**6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular (1.000 carateres).**

Esta unidade curricular representa, para a maioria dos alunos, o primeiro contacto com as problemáticas inerentes ao desenvolvimento de aplicações distribuídas, concretizadas sobre a plataforma Web. São introduzidos os principais elementos constituintes desta plataforma (I a III do conteúdo programático). Apresenta-se a componente de cliente (web browser) e as respetivas tecnologias normalizadas que lhe estão associadas (IV). Descreve-se a componente de servidor, identificando os principais padrões arquiteturais usados no seu desenvolvimento, demonstrados na prática através de uma infraestrutura de servidor com significativa adoção industrial (V). O desenvolvimento de uma aplicação Web de pequena ou média complexidade é usada para praticar os conceitos e tecnologias transmitidas e colocar os alunos perante alguns os problemas associados ao desenvolvimento deste tipo de aplicações (VI e VII).

**6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes (1.000 characters).**

For most students, this course represents the first contact with the problems inherent to the development of distributed applications, achieved through the web platform. The main elements of this platform are introduced (I to III of the syllabus). The client component (web browser) is presented, with its associated standardized technologies (IV). The server component is also described, identifying the key architectural patterns used in its development, which are demonstrated in practice through a web infrastructure with significant industrial adoption (V). The development of a web application with small or medium complexity is used to practice the transmitted concepts and technologies and to put students before some of the problems associated with the development this type of applications (VI and VII).

**7. Metodologias de ensino (avaliação incluída) (1.000 carateres).**

Ensino teórico-prático, com 30 aulas (15 aulas de 3 horas e 15 de 1,5 horas). As aulas destinam-se à apresentação e explicação dos temas e à demonstração da sua aplicação prática. Os tópicos principais são ainda explorados através da realização de uma aplicação Web ao longo do semestre, por grupos de 2 ou 3 alunos. O desenvolvimento desta aplicação é dividido em 3 etapas, cada uma introduzindo novos requisitos, incorporando assim os temas que vão sendo apresentados.

Os resultados (1) a (6) são avaliados individualmente através do teste escrito realizado no final do semestre (T) e na avaliação de cada etapa do trabalho prático realizado durante o semestre (P). Durante o acompanhamento dos trabalhos de grupo realizados nas aulas práticas são avaliados os resultados (6) e (7). Todos os resultados da aprendizagem são também avaliados na discussão final dos trabalhos de grupo. A nota final é atribuída segundo a seguinte fórmula:  $40\%*T + 60\%*P$ .

**7. Teaching methodologies (including assessment) (1.000 characters).**

Theoretical and practical teaching in 30 lectures (15 lessons of 3 hours and 15 1.5 hours). The lectures are intended for presentation and explanation of the topics and their practical application demonstration. The main topics are further explored by developing a web application throughout the semester, in groups of 2 or 3 students. The application development is divided in 3 stages, each one introducing new requirements to incorporate the subjects being presented.

Learning outcomes (1) to (6) are evaluated individually through the written exam conducted at the end of the semester (E), and through the evaluation of each stage of the practical work delivered during the semester (P). During follow-up of group work performed in practical lectures, the learning outcomes (6) and (7) are also assessed. All learning outcomes are also evaluated in the group final discussion of work group. The final grade is assigned according to the following formula:  $40\%*E + 60\%*P$ .

**8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular (3.000 carateres).**

Nas sessões teóricas são apresentados os conceitos fundamentais para o desenvolvimento de aplicações para a Web. Nas sessões práticas, estes conceitos são complementados através de exemplos relacionados com o trabalho, praticados e consolidados no seu desenvolvimento. A realização das etapas do trabalho tem uma complexidade gradualmente crescente, de modo a incorporar os novos temas que vão sendo tratados ao longo do semestre. Em cada etapa é criada documentação que fica disponível online, integrada no sistema de controlo de versões usado. Esta documentação inclui as funcionalidades implementadas e decisões tomadas em cada etapa. Na etapa final é criada documentação no mesmo suporte na forma de relatório final, que descreve a totalidade do trabalho desenvolvido, a sua arquitetura, decisões de implementação e respetivas fundamentações.

**8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes (3.000 characters).**

In theoretical sessions the basic concepts for developing web applications are presented. In practice sessions, these concepts are complemented with examples related to the practical work, practiced and consolidated in its development. The stages of the practical work gradually increase complexity, in order to incorporate the new themes addressed throughout the semester. At each stage online documentation is made available, incorporated in the version control system used. This documentation includes the implemented features and decisions made at each stage. In the final stage, documentation is created in the form of a final report, which describes the work entirely, its architecture, implementation decisions and their corresponding rationales.

**9. Bibliografia de consulta/existência obrigatória (1.000 carateres).**

- W3C Technical Architecture Group, Architecture of the World Wide Web, Volume One, <http://www.w3.org/TR/webarch/>, 2004.
- Hypertext Transfer Protocol -- HTTP/1.1, RFC 2616.
- M. Haverbeke, *Eloquent JavaScript: A Modern Introduction to Programming*, No Starch Press, 2011. ISBN 9781593272821, <http://eloquentjavascript.net>
- J. Wilson, *Node.js 8 the Right Way: Practical, Server-Side JavaScript That Scales*, The Pragmatic Bookshelf, 2017. ISBN 9781937785734

<sup>1</sup> Anual, semestral, trimestral, ...

<sup>2</sup> Número total de horas de trabalho.

<sup>3</sup> Discriminadas por tipo de metodologia adotado (T - Ensino teórico; TP - Ensino teórico-prático; PL - Ensino prático e laboratorial; TC - Trabalho de

campo; S - Seminário; E - Estágio; OT - Orientação tutorial; O - Outro).

<sup>4</sup> Assinalar sempre que a unidade curricular seja optativa.