

## Ficha da unidade curricular (versão A3ES)

### 6.2.1.1. Unidade curricular

Programação na Internet / Internet Programming

### 6.2.1.2. Docente responsável e respetivas horas de contacto na unidade curricular (preencher o nome completo)

José Luís Falcão Cascalheira, 67,5 horas de contacto

### 6.2.1.3. Outros docentes e respetivas horas de contacto na unidade curricular

Fernando Miguel Santos Lopes de Carvalho, 67,5 horas de contacto

### Other academic staff and lecturing load in the curricular unit

Fernando Miguel Santos Lopes de Carvalho, 67.5 contact hours

### 6.2.1.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes)

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

1. Conhecer os principais elementos constituintes da arquitectura da *World Wide Web*
2. Compreender o protocolo HTTP
3. Demonstrar conhecimento teórico-prático sobre as principais normas associadas à componente de cliente (web browser)
4. Conhecer os principais padrões de apresentação utilizados na componente de servidor
5. Compreender, utilizar e estender uma tecnologia de servidor, com significativa adopção industrial, para criação de aplicações Web
6. Desenhar e implementar aplicações Web de pequena e média complexidade, que incluem funcionalidades de autenticação, manutenção de estado de conversação, visualização e edição de dados
7. Desenvolvimento de software em grupo, utilizando como suporte um sistema de controlo de versões distribuído, para manter os artefactos de software desenvolvidos e produzir documentação

### Learning outcomes of the curricular unit

Students who successfully complete this course unit be able to:

1. Know the main components of the of the World Wide Web architecture
2. Understand the HTTP protocol
3. Demonstrate theoretical and practical knowledge on key standards associated with the client component (web browser)
4. Know the main presentation patterns used in server component
5. Understand, use and extend a server technology, with significant industrial adoption, for creating web applications
6. Design and implement Web applications for small and medium complexity, including authentication features, maintenance of conversation state, viewing and editing data
7. Software development is group, using a distributed version control system for its support, to maintain the produced software artifacts and its corresponding documentation

### 6.2.1.5. Conteúdos programáticos

- I. A arquitectura da World Wide Web: Identificação de recursos (URIs), interacção (protocolo HTTP) e representação (HTML).
- II. Distribuição de conteúdos web (protocolo HTTP).
- III. Infra-estruturas de suporte à criação da interface com o utilizador em aplicações web e respectivo modelo de programação.
- IV. Componente de cliente (Web browser): descrição, formatação visual, manipulação programática e actualizações 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); intercepção de pedidos.
- VI. Consequências da distribuição no modelo de programação e respectivas arquitecturas de referência. Mecanismos de cache.
- VII. Concretização dos temas estudados através do desenvolvimento de aplicações Web de pequena e média complexidade com suporte para persistência de dados.

## Syllabus

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 (ex. Relational Database).

### 6.2.1.6. Demonstração da coerência dos conteúdos programáticos com os objetivos da unidade curricular

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 respectivas 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 infra-estrutura 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).

### Demonstration of the syllabus coherence with the curricular unit's objectives

This course represents, for most students, 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).

### 6.2.1.7. Metodologias de ensino (avaliação incluída)

Ensino teórico-prático, estando previstas 30 aulas durante o semestre a que correspondem 67,5 horas de contacto (15 aulas de 3 horas e 15 de 1,5 horas). O tempo total de trabalho do estudante é de 160 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, de modo a incorporar os temas que vão sendo apresentados.

Os resultados (1) a (6) são avaliados individualmente através do teste escrito realizado no final do semestre e na avaliação de cada etapa do trabalho durante o semestre. 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.

### Teaching methodologies (including evaluation)

Theoretical and practical teaching is planned during the semester in 30 lectures that correspond to 67.5 of contact hours (15 lessons of 3 hours and 15 1.5 hours). The total student working hours is 160. 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 test conducted at the end of the semester, and through the evaluation of each stage of the practical work delivered during the

semester. 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.

#### **6.2.1.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular**

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 respectivas fundamentações.

#### **Demonstration of the coherence between the teaching methodologies and the learning outcomes**

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.

#### **6.2.1.9. Bibliografia principal**

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.

Marijn Haverbeke, Eloquent JavaScript, No Starch Press (<http://eloquentjavascript.net>), 2011.

Steven Sanderson , Adam Freeman, Pro ASP.NET MVC 3 Framework, APress, 2011.