
1. Caracterização da Unidade Curricular

1.1 Designação

[4167] Introdução à Programação na Web / Introduction to Internet Programming

1.2 Sigla da área científica em que se insere

IC

1.3 Duração

Unidade Curricular Semestral

1.4 Horas de trabalho

162h 00m

1.5 Horas de contacto

Total: 67h 30m das quais TP: 67h 30m

1.6 ECTS

6

1.7 Observações

Unidade Curricular Obrigatória

2. Docente responsável

[1245] José Luís Falcão Cascalheira

3. Docentes e respetivas cargas letivas na unidade curricular

[1245] José Luís Falcão Cascalheira | Horas Previstas: 135 horas

[1604] Filipe Bastos de Freitas | Horas Previstas: 67.5 horas

[2108] Fernanda Gonçalves de Oliveira Passos | Horas Previstas: 135 horas

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. Enquadrar os principais elementos constituintes da arquitetura da World Wide Web;
2. Desenhar e consumir Web API que seguem os princípios REST.
3. Demonstrar conhecimento teórico-prático sobre as principais normas associadas à componente de cliente (web browser);
4. Utilizar o modelo de programação na componente cliente, nomeadamente no acesso dinâmico a conteúdos e tratamento de eventos;
5. Utilizar e estender uma tecnologia de servidor com significativa adoção industrial, para criação de aplicações Web de pequena complexidade;
6. 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)**

Students who successfully complete this course unit be able to:

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

5. Conteúdos programáticos

1. A arquitetura da *World Wide Web* : Identificação de recursos (URIs), interação (protocolo HTTP) e representação.
2. Distribuição de conteúdos web (protocolo HTTP).
3. Criação e utilização de REST APIs.
4. Infraestruturas de suporte à criação da interface com o utilizador em aplicações web e respetivo modelo de programação.
5. Componente de cliente (Web browser): descrição da estrutura (HTML), formatação visual (CSS), manipulação programática e atualizações totais e parciais da interface gráfica.
6. Componente de servidor: distribuição de conteúdos estáticos; geração dinâmica de conteúdos; modelo de programação no servidor, manutenção de estado (de visualização, de sessão e de aplicação); interceção de pedidos.
7. Consequências da distribuição no modelo de programação e respetivas arquiteturas de referência.
8. Concretização dos temas estudados através do desenvolvimento de aplicações Web de pequena e média complexidade com suporte para persistência de dados.

5. Syllabus

1. Architecture of the World Wide Web: Resource Identification (URI), interaction (HTTP) and representation (HTML).
2. Distribution of web content (HTTP protocol).
3. Creation and consumption of REST APIs.
4. Support Infrastructures for the creation of user interface in web applications and their programming model.
5. Client Component (web browser): description, visual formatting, programmatic manipulation and total and partial updates of the GUI.
6. Server component: static content distribution, dynamic generation of content; programming model on the server, maintaining state (view, session and application); HTTP requests intercepting.
7. Distribution consequences in the programming model and corresponding reference architectures. Caching mechanisms.
8. 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

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 IV do conteúdo programático). Apresenta-se a componente de cliente (web browser) e as respetivas tecnologias normalizadas que lhe estão associadas (V). 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 (VI). 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 dos problemas associados ao desenvolvimento deste tipo de aplicações (VII e VIII).

6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes

This curricular unit represents, for most students, the first contact with the problems inherent to the development of distributed applications, implemented on the Web platform. The main constituent elements of this platform are introduced (I to IV of the syllabus). The client component (web browser) and the respective standardized technologies associated with it are presented (V). The server component is described, identifying the main architectural patterns used in its development, demonstrated in practice through a server infrastructure with significant industrial adoption (VI). The development of a Web application of small or medium complexity is used to practice the concepts and technologies transmitted and expose students to some of the problems associated with the development of this type of applications (VII and VIII).

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

Ensino teórico-prático com 30 aulas (15 de 1,5h e 15 de 3h), correspondentes a 67,5 horas de contacto. O tempo total de trabalho do estudante é de 162 horas. As aulas destinam-se à apresentação e explicação dos temas e à sua demonstração prática.

A unidade curricular usa avaliação distribuída com exame final. Durante o período letivo, todos os tópicos do programa são explorados através da realização de trabalhos práticos (CP), com classificação individual atribuída em discussões orais. A CP inclui uma série de exercícios individual (E) e uma Aplicação Web (A) pedagogicamente fundamental, desenvolvida em 3 etapas ao longo do semestre, por grupos de 2 ou 3 alunos. No final do período letivo realiza-se um exame escrito (T).

Todos os resultados da aprendizagem são avaliados individualmente através de T e CP e na discussão final oral. A nota final é atribuída segundo a seguinte fórmula: $40\%*T + 60\%*CP$, onde CP é E+A. Todas avaliações têm um mínimo de 9,5 valores.

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

Theoretical and practical teaching with 30 lectures (15 of 1.5 hours and 15 of 3 hours), corresponding to 67.5 contact hours. The student's total work time is 162 hours. The classrooms are intended to present and explain the syllabus topics and their practical demonstration.

The curricular unit uses distributed assessment with a final exam. During the teaching period, all topics are explored through practical work (CP), with individual grades attributes in oral discussions. The CP includes a series of individual exercises (E) and a pedagogically fundamental Web Application (A), developed in 3 stages throughout the semester, by groups of 2 or 3 students. At the end of the school period, a written exam (T) is carried out.

All learning results are assessed individually through T and CP and in a final oral discussion. The final grade is attributed according to the following formula: $40\%*T + 60\%*CP$, where CP is determined by E and A. All endorsements have a minimum value of 9.5, for approval.

**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 respetivas fundamentações.

8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes

In theoretical sessions, the fundamental concepts for developing web applications are presented. In practical sessions, these concepts are complemented through work-related examples, practiced and consolidated in their development. Carrying out the stages of the work has a gradually increasing complexity, in order to incorporate the new themes that will be addressed throughout the semester. At each stage, documentation is created and made available online, integrated into the version control system used. This documentation includes the implemented features and decisions made at each stage. In the final stage, documentation is created on the same medium in the form of a final report, which describes the entire work developed, its architecture, implementation decisions and respective justifications.

9. Bibliografia de consulta/existência obrigatória

- 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>
- Shay Howe, Learn to Code HTML and CSS: Develop and Style Websites, New Riders, May 2014. ISBN-10: 0-321-94052-0, <https://learn.shayhowe.com/html-css>
- J. Wilson, *Node.js 8 the Right Way: Practical, Server-Side JavaScript That Scales*, The Pragmatic Bookshelf, 2017. ISBN 9781937785734

10. Data de aprovação em CTC 2024-07-17

11. Data de aprovação em CP 2024-06-26