
1. Designação da unidade curricular

[2641] Laboratório de Software / Software Laboratory

2. Sigla da área científica em que se insere

IC

3. Duração

Unidade Curricular Semestral

4. Horas de trabalho

162h 00m

5. Horas de contacto

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

6. % Horas de contacto a distância

Sem horas de contacto à distância

7. ECTS

6

8. Docente responsável e respetiva carga letiva na Unidade Curricular

[1604] Filipe Bastos de Freitas | Horas Previstas: N/D

9. Outros docentes e respetivas cargas letivas na unidade curricular

Não existem docentes definidos para esta unidade curricular

10. Objetivos de aprendizagem e a sua compatibilidade com o método de ensino (conhecimentos, aptidões e competências a desenvolver pelos estudantes).

1. Desenvolver software aplicacional utilizando sistemas de gestão de bases de dados e interfaces Web, incluindo todos os elementos de apoio à sua manutenção e evolução, nomeadamente documentação técnica;
2. Usar ferramentas de suporte ao desenvolvimento de projetos de software, nomeadamente controlo de versões, automatização de tarefas de construção e testes unitários de software;
3. Usar metodologias de desenvolvimento ágeis, enquadradas no trabalho em equipas desenvolvimento de software em projetos de média dimensão;
4. Escrever um relatório técnico a descrever e enquadrar o trabalho realizado.

10. Intended Learning objectives and their compatibility with the teaching method (knowledge, skills and competences by the students).

1. Develop software systems using relational database management systems and Web interfaces, including all the supporting elements for its maintenance and evolution, namely documentation.
2. Correctly use software development tools, such as version control systems, build automation tools and testing frameworks.
3. Use agile development methodologies, aligned with the work in software development teams of medium-sized projects;
4. Write a technical report describing and framing the work performed.

11. Conteúdos programáticos

1. Sistemas de Controlo de Versões e formas de utilização (e.g. Git).
2. Organização do ambiente de desenvolvimento e processos de automatização no processo de construção de aplicações - "build automation" (e.g. Gradle).
3. Testes não supervisionados (unitários, funcionais e de integração) e plataformas de suporte (e.g. JUnit).
4. Utilização de aplicações de suporte ao desenvolvimento de projetos, nomeadamente nas componentes de controlo de versões, wiki e issue tracking (e.g. GitHub).

11. Syllabus

1. Version control systems and their associated usage practices (e.g. Git).
2. Development environment organization and build automation processes (e.g. Gradle).
3. Automatic software tests (unitary, functional and integration) and associated frameworks (e.g. JUnit).
4. Development supporting applications, namely wikis and issue trackers (e.g. GitHub).

12. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular

Esta unidade curricular tem como primeiro objetivo a consolidação dos conceitos e práticas fornecidos aos estudantes em unidades curriculares anteriores, nas áreas da programação e sistemas de informação, bem como melhorar a capacidade de escrita de relatórios técnicos.

Assim, o ponto 1. dos objetivos é assegurado através da utilização e aprofundamento de conhecimentos fornecidos nestas unidades curriculares anteriores.

O segundo propósito desta unidade curricular é dotar os estudantes das competências necessários ao trabalho em grupo (objetivo 3) e à utilização adequada de métodos e ferramentas utilizadas no desenvolvimento profissional de software (objetivo 2 e 3). Os conteúdos 1, 2, 3 e 4, bem como a sua utilização aplicada no contexto do desenvolvimento dum projeto, contribuem para estes objetivos. A escrita do relatório técnico sobre o trabalho desenvolvido contribuem para o objetivo 4

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

This course unit aims firstly to consolidate the concepts and practices provided to students in previous courses in the areas of programming and information systems, as well as to improve the ability to write technical reports. Thus, objective 1 is ensured through the use and deepening of knowledge provided in these previous courses units. The second purpose of this course unit is to give students the skills necessary to work in group (objective 3) using the appropriate methods and tools used in professional software development (objectives 2 and 3). Contents 1, 2, 3, and 4, as well as their applied use in the context of project development, contribute to these objectives. The writing of a technical report on the work developed contributes to objective 4.

13. Metodologias de ensino e de aprendizagem específicas da unidade curricular articuladas com o modelo pedagógico

Ensino realizado em ambiente de prática laboratorial (30 aulas, 67,5 horas de contacto). O tempo total de trabalho do estudante é de 162 horas. As primeiras 2 semanas são usadas para a apresentação teórica dos conteúdos programáticos. Uma semana é usada para a avaliação individual e apresentação do trabalho realizado. As restantes 12 semanas são usadas no desenvolvimento iterativo de um projeto de software, dividido em quatro etapas de 3 semanas cada e realizado em grupos de 2 a 3 alunos. Durante estas, as horas de contacto são usadas para orientar os alunos no projeto, bem como para apresentar ou complementar temas necessários à sua realização.

A avaliação é distribuída sem exame final, a avaliação do projeto é realizada individualmente com cada grupo de trabalho, tendo em conta todos os elementos produzidos.

A classificação final (CF) de cada aluno é baseada na entrega do projeto e na sua discussão. CF = 100% discussão do projeto.

13. Teaching and learning methodologies specific to the curricular unit articulated with the pedagogical model

This unit is taught in a laboratory environment (30 lectures, 67.5 contact hours). The students are expected to use a total of 162 hours. The first 2 weeks are used for the theoretical presentation of the syllabus contents. One week is used for individual student evaluations as well as for the students to showcase the work done. The remaining 12 weeks are used in the iterative development of a software project, divided into 4 stages with 3 weeks each. During these 12 weeks, the contacts hours are used primarily to supervise the students; as well as to present or complement additional topics required for the projects. This activity is done in a laboratory environment. The evaluation is distributed without final exam. The project evaluation is performed individually for each student group. The final student grade is based on the project delivery and its discussion. FG = 100% project discussion.

14. Avaliação

Ensino realizado em ambiente de prática laboratorial (30 aulas, 67,5 horas de contacto). O tempo total de trabalho do estudante é de 162 horas. As primeiras 2 semanas são usadas para a apresentação teórica dos conteúdos programáticos. Uma semana é usada para a avaliação individual e apresentação do trabalho realizado. As restantes 12 semanas são usadas no desenvolvimento iterativo de um projeto de software, dividido em quatro etapas de 3 semanas cada e realizado em grupos de 2 a 3 alunos. Durante estas, as horas de contacto são usadas para orientar os alunos no projeto, bem como para apresentar ou complementar temas necessários à sua realização. A avaliação é distribuída sem exame final, a avaliação do projeto é realizada individualmente com cada grupo de trabalho, tendo em conta todos os elementos produzidos. A classificação final (CF) de cada aluno é baseada na entrega do projeto e na sua discussão. CF = 100% discussão do projeto.

14. Assessment

This unit is taught in a laboratory environment (30 lectures, 67.5 contact hours). The students are expected to use a total of 162 hours. The first 2 weeks are used for the theoretical presentation of the syllabus contents. One week is used for individual student evaluations as well as for the students to showcase the work done. The remaining 12 weeks are used in the iterative development of a software project, divided into 4 stages with 3 weeks each. During these 12 weeks, the contacts hours are used primarily to supervise the students; as well as to present or complement additional topics required for the projects. This activity is done in a laboratory environment. The evaluation is distributed without final exam. The project evaluation is performed individually for each student group. The final student grade is based on the project delivery and its discussion. FG = 100% project discussion.

15. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular

O ensino desta unidade curricular é maioritariamente realizado em ambiente laboratorial, consistindo no esclarecimento e aconselhamento dos alunos sobre os desafios do projeto e da sua organização. A existência de quatro etapas, com entrega de versões funcionais do projeto por parte dos grupos de trabalho, proporciona a existência de vários momentos de avaliação. Nestes, os docentes fazem uma análise crítica do trabalho já realizado e perspetivam melhoramentos.

Considera-se que este ensino e avaliação iterativa, baseada em trabalho realizado pelos alunos, é adequado aos objetivos da unidade curricular, nomeadamente os pontos 1 e 2. A realização em grupo de um projeto com a duração de 12 semanas e várias versões é adequada ao objetivo 3. A documentação produzida é adequada ao objetivo 4

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

The teaching of this course unit is primarily conducted in a laboratory environment, consisting of clarifying and advising students on the challenges of the project and its organization. The existence of four stages, with the submission of functional versions of the project by the working groups, provides several evaluation moments. During these, teachers perform a critical analysis of the work already done and suggest improvements. It is considered that this iterative teaching and evaluation, based on the work carried out by the students, is appropriate for the objectives of the course unit, namely objectives 1 and 2. The group execution of a 12-week project with multiple versions is suitable for objective 3. The documentation produced is appropriate for objective 4.

**16. Bibliografia de
consulta/existência obrigatória**

Conjunto de notas disponibilizadas em repositório público (e.g. <https://github.com/isel-leic-ls/2122-2-common/wiki>), incluindo referências para conteúdos externos, também de acesso público (e.g. S. Chacon, B. Straub, ?Pro Git?, Apress, 2018).

Bibliografia das unidades curriculares anteriores, nas áreas da programação e dos sistemas de informação

17. Observações

Unidade Curricular Obrigatória

Data de aprovação em CTC:

Data de aprovação em CP: