# NCE/15/00113 — Apresentação do pedido - Novo ciclo de estudos

## Apresentação do pedido

### Perguntas A1 a A4

**A1. Instituição de ensino superior / Entidade instituidora:**
***Universidade Do Porto***

**A1.a. Outras Instituições de ensino superior / Entidades instituidoras:**

**A2. Unidade(s) orgânica(s) (faculdade, escola, instituto, etc.):**
***Faculdade De Engenharia (UP)***

**A3. Designação do ciclo de estudos:**
***Engenharia de Software***

**A3. Study programme name:**
***Software Engineering***

**A4. Grau:**
***Mestre***

### Perguntas A5 a A10

**A5. Área científica predominante do ciclo de estudos:**
***Ciências Informáticas***

**A5. Main scientific area of the study programme:**
***Computer Science***

**A6.1. Classificação da área principal do ciclo de estudos (3 dígitos), de acordo com a Portaria n.º 256/2005, de 16 de Março (CNAEF):**
***481***

**A6.2. Classificação da área secundária do ciclo de estudos (3 dígitos), de acordo com a Portaria n.º 256/2005, de 16 de Março (CNAEF), se aplicável:**
***N/A***

**A6.3. Classificação de outra área secundária do ciclo de estudos (3 dígitos), de acordo com a Portaria n.º 256/2005, de 16 de Março (CNAEF), se aplicável:**
***N/A***

**A7. Número de créditos ECTS necessário à obtenção do grau:**
***120***

**A8. Duração do ciclo de estudos (art.º 3 DL-74/2006, de 26 de Março):**
***4 semestres***

**A8. Duration of the study programme (art.º 3 DL-74/2006, March 26th):**
***4 semesters***

**A9. Número de vagas proposto:**
***30***

**A10. Condições especificas de ingresso:**
***Podem candidatar-se ao Mestrado em Engenharia de Software:***
***• Titulares do grau de licenciado ou equivalente legal em Engenharia Informática, Ciências de Computadores, ou similar.***
***• Titulares de um grau académico superior estrangeiro conferido na sequência de um 1.º ciclo de estudos em área adequada (Ciência de Computadores, Engenharia Informática, ou similar) organizado de acordo com o Processo de Bolonha por um Estado aderente a este Processo.***
***• Titulares de um grau académico superior estrangeiro em área adequada (Ciência de Computadores, Engenharia Informática, ou similar) que seja reconhecido como satisfazendo os objetivos do grau de licenciado pelo órgão científico estatutariamente competente;***
***• Detentores de um currículo escolar, científico ou profissional que seja reconhecido como atestando capacidade para realização deste ciclo de estudos pelo órgão científico estatutariamente competente;***
***• É requerida proficiência comprovada em Inglês.***

**A10. Specific entry requirements:**
***Eligible applicants for the Masters in Software Engineering must fulfil one of the following requirements:***
***• BSc degree, or legal equivalent, in Informatics, Computer Science or related area.***
***• Higher education academic degree taken abroad, in a suitable area (Informatics Engineering, Computer Science, or equivalent), corresponding to a 1st stage study organised according to the Bologna Process by a State subscribing to this Process.***
***• Higher education academic degree taken abroad, in a suitable area (Informatics Engineering, Computer Science, or equivalent), recognised, by the hosting academic institution, as fulfilling of the requirements of a BSc degree.***
***• Applicants holding a scientific or professional academic curriculum that provides evidence of the skills required to undertake this course, and is recognised by the legally competent scientific body of the hosting institution.***
***• Besides one of these, proven proficiency in English is required.***

## Pergunta A11

**Pergunta A11**

**A11. Percursos alternativos como ramos, variantes, áreas de especialização do mestrado ou especialidades do doutoramento em que o ciclo de estudos se estrutura (se aplicável):**
***Não***

**A11.1. Ramos, variantes, áreas de especialização do mestrado ou especialidades do doutoramento (se aplicável)**

**A11.1. Ramos, variantes, áreas de especialização do mestrado ou especialidades do doutoramento, em que o ciclo de estudos se estrutura (se aplicável) / Branches, options, specialization areas of the master or specialities of the PhD (if applicable)**

| Ramo, variante, área de especialização do mestrado ou especialidade do doutoramento: | Branch, option, specialization area of the master or speciality of the PhD: |
|---|---|
| ***<sem resposta>*** | |

## A12. Estrutura curricular

**Mapa I - Não aplicável**

**A12.1. Ciclo de Estudos:**
***Engenharia de Software***

**A12.1. Study Programme:**
***Software Engineering***

**A12.2. Grau:**

***Mestre***

**A12.3. Ramo, variante, área de especialização do mestrado ou especialidade do doutoramento (se aplicável):**
***Não aplicável***

**A12.3. Branch, option, specialization area of the master or speciality of the PhD (if applicable):**
***<no answer>***

**A12.4. Áreas científicas e créditos que devem ser reunidos para a obtenção do grau / Scientific areas and credits that must be obtained for the awarding of the degree**

| Área Científica / Scientific Area | Sigla / Acronym | ECTS Obrigatórios / Mandatory ECTS | ECTS Minímos Optativos* / Minimum Optional ECTS* |
|---|---|---|---|
| Ciências Informáticas/ Computer Science | CINF | 108 | 0 |
| Ciências Informáticas/Qualquer área científica da UPorto(ao nível do 2º ciclo)/ Computer Science/any Scientic area from UPorto (2nd cycle) | CINF/QACUP | 0 | 12 |
| **(2 Items)** | | **108** | **12** |

## Perguntas A13 e A16

**A13. Regime de funcionamento:**
***Diurno***

**A13.1. Se outro, especifique:**
***Não aplicável.***

**A13.1. If other, specify:**
***Not applicable.***

**A14. Local onde o ciclo de estudos será ministrado:**
***Faculdade de Engenharia da Universidade do Porto.***

**A14. Premises where the study programme will be lectured:**
***Faculty of Engineering of the University of Porto***

**A15. Regulamento de creditação de formação e experiência profissional (PDF, máx. 500kB):**
**A15._Regulamento Creditação Formação Anterior Experiência ProfissionalUP.pdf**

**A16. Observações:**
***A Engenharia de Software(ES)concentra-se na aplicação d abordagens sistemáticas,disciplinadas e quantificáveis para o desenvolvimento,operação e manutenção d sistemas d software.A ES preocupa-se com metodologias,técnicas e ferramentas para gerir todo o ciclo d vida do software,incluindo o levantamento d requisitos,especificação,desenho,testes,manutenção e gestão d projetos.A ES resulta da procura contínua d qualidade de software e o amadurecimento da indústria d desenvolvimento d software.***
***O Mestrado em Engenharia de Software (MESW) tem por objetivo oferecer formação científica e profissional avançada em ES e é projetado para profissionais q buscam atualizar as suas competências,bem como,para aqueles que pretendem adquirir novas competências e conhecimentos atuais em ES.Durante este programa,os estudantes irão adquirir conhecimentos teóricos e competências para lidar com a complexidade d sistemas d software,bem como,compreender e desenvolver sistemas d software intensivos.Os estudantes irão adquirir conhecimentos para analisar,desenhar,implementar,testar,compreender e manter/evoluir sistemas de software grandes e complexos.O curriculum do MESW está ancorado em problemas do mundo real com uma componente d projetos d desenvolvimento em equipa para"aprender fazendo".O MESW visa assim formar profissionais altamente especializados capazes de assumir a liderança d projetos d engªd software complexos e d grande dimensão com exigências d qualidade.***
***O plano curricular do MESW permite que os estudantes adquiram,entre outras,competências nas seguintes áreas:***
***-Fundamentos e paradigmas em ES***
***-Especificação,desenho e construção de software***
***-Questões d segurança em ES***
***-Compreensão,evolução,teste,validação e verificação de software***
***-Inovação e empreendedorismo em ES***

***Os graduados do MESW são profissionais absolutamente indispensáveis à indústria de software,capazes d gerir projetos d grande dimensão,complexidade e restrições impostas pelo mercado.Portugal precisa de ser capaz d formar as pessoas capazes de estar no topo da hierarquia técnica da indústria de software mundial.***
***O MESW pretende ser um CE transfronteiriço,procurando parcerias internacionais que fomentem a circulação de estudantes,docentes e investigadores e construindo ligações com entidades formadoras e grupos d investigação internacionais.***
***O MESW inclui uma componente curricular de 72 ECTS (cursos de mestrado–não conferente de grau) e uma Dissertação de 48 ECTS.Na componente curricular,60 ECTS são afetos a 10 UC's obrigatórias e 12 ECTS são afetos a 2 opcionais (uma no 1ºano/2ºS, e outra no 2ºano/1ºS).***
***Tendo em vista a internacionalização,pretende-se que este CE seja lecionado em Inglês.***
***Relativamente ao campo 2.5.(1º ano/2ºS e 2º ano) efetuamos a seguinte observação:Pode,o Diretor da FEUP autorizar, mediante parecer positivo do Conselho Científico e sob proposta da Comissão Científica do CE,a alteração da distribuição das UC's optativas pelos semestres num determinado ano lectivo.***

**A16. Observations:**

***The Software Engineering focuses on the application of systematic, disciplined and quantifiable approaches for the development, operation and maintenance of software systems. Its main concerns address the methodologies, techniques and tools that allow the management of the whole software lifecycle, namely requirements elicitation and specification, design, implementation, testing, maintenance and project management. This area arises from the need for the continuous improvement of the quality of software and the maturing of the software development industry.***

***The Master of Science in Software Engineering (MESW) aims at providing advanced, professional and scientific training, devised for professionals seeking to update their skills, as well as to acquire new, state-or-the-art knowledge in Software Engineering. During the course of the programme, students will acquire sound theoretical knowledge, enabling them to handle complex software systems, besides understanding and developing software intensive systems. Students will become proficient in analysis, design, implementation, testing, understanding and maintaining/evolving large and complex software systems. The MESW curriculum is anchored in real world problem solving, through team based software development projects, based on a "learn-by-doing" philosophy. Consequently, MESW provides the industry with highly trained, specialised professionals capable of leading and managing large and complex software engineering projects, guaranteeing high quality standards.***

***MESW study plan covers the following areas:***
***- SE fundamentals and paradigms;***
***- Software specification, design and construction;***
***- Security issues in SE;***
***- Software understanding, evolution, testing and verification and validation.***
***- Innovation and entrepreneurship in SE.***

***MESW graduates become indispensable professionals to the software industry, coping with size, complexity and market needs and constraints. Portugal needs to able to place its trained professionals at the top of the world's software industry technical hierarchy. MESW intends to be a transnational study cycle, establishing international partnerships that promote mobility for students, teachers and researchers, while building bonds with international training institutions and research groups.***
***MESW is a two-year course with a 72 ECTS curricular component and a final dissertation worth 48 ECTS. Of the 72, 60 ECTS cover 10 mandatory modules, while 12 ECTS cover two optional modules (1st year, 2nd semester and 2nd year, 1st semester). The official language of MESW is English.***
***In relation to question 2.5 (1 year / 2ºS and 2nd year): Can the Director of FEUP authorize, upon favorable opinion of the Scientific Council and proposal from the Scientific Committee study programme, to change the distribution of UC's optional by semesters in a given academic year.***

# Instrução do pedido

## 1. Formalização do pedido

### 1.1. Deliberações

**Mapa II - Conselho Científico da Faculdade de Engenharia da Universidade do Porto**

**1.1.1. Órgão ouvido:**
***Conselho Científico da Faculdade de Engenharia da Universidade do Porto***

**1.1.2. Cópia de acta (ou extrato de acta) ou deliberação deste orgão assinada e datada (PDF, máx. 100kB):**
**1.1.2._MESW CC FEUP.pdf**

**Mapa II - Conselho Pedagógico da Faculdade de Engenharia da Universidade do Porto**

**1.1.1. Órgão ouvido:**
***Conselho Pedagógico da Faculdade de Engenharia da Universidade do Porto***

**1.1.2. Cópia de acta (ou extrato de acta) ou deliberação deste orgão assinada e datada (PDF, máx. 100kB):**
**1.1.2._MESW CP FEUP.pdf**

**Mapa II - Reitor da Universidade do Porto**

**1.1.1. Órgão ouvido:**
***Reitor da Universidade do Porto***

**1.1.2. Cópia de acta (ou extrato de acta) ou deliberação deste orgão assinada e datada (PDF, máx. 100kB):**
**1.1.2._DespachoReitoralCriacao_2C_EngenhariaSoftware.pdf**

**1.2. Docente(s) responsável(eis) pela coordenação da implementação do ciclo de estudos**

**1.2. Docente(s) responsável(eis) pela coordenação da implementação do ciclo de estudos**
**A(s) respectiva(s) ficha(s) curricular(es) deve(m) ser apresentada(s) no Mapa V.**
***Diretor do ciclo de estudos: Ana Cristina Ramada Paiva (Professora Auxiliar)***

## 2. Plano de estudos

**Mapa III - - 1º ano / 1º semestre**

**2.1. Ciclo de Estudos:**
***Engenharia de Software***

**2.1. Study Programme:**
***Software Engineering***

**2.2. Grau:**
***Mestre***

**2.3. Ramo, variante, área de especialização do mestrado ou especialidade do doutoramento (se aplicável):**
***<sem resposta>***

**2.3. Branch, option, specialization area of the master or speciality of the PhD (if applicable):**
***<no answer>***

**2.4. Ano/semestre/trimestre curricular:**
***1º ano / 1º semestre***

**2.4. Curricular year/semester/trimester:**
***1st year / 1st semester***

**2.5. Plano de Estudos / Study plan**

| Unidade Curricular / Curricular Unit | Área Científica / Scientific Area (1) | Duração / Duration (2) | Horas Trabalho / Working Hours (3) | Horas Contacto / Contact Hours (4) | ECTS | Observações / Observations (5) |
|---|---|---|---|---|---|---|
| Princípios e Paradigmas de Engenharia de Software/Software Engineering Fundamentals and Paradigms | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Engenharia de Requisitos e Modelação de Software/ Requirements Engineering and Software Modeling | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Arquitetura e Desenho de Software/Software Architecture and Design | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |

| | | | | | | |
|---|---|---|---|---|---|---|
| Teste, Verificação e Validação de Software/Software Testing Verification and Validation | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Laboratório de Engenharia de Software/Software Engineering Laboratory | CINF | Semestral | 162 | 14 TP + 28 PL | 6 | Obrigatória |

**(5 Items)**

**Mapa III - - 1º ano / 2º semestre**

**2.1. Ciclo de Estudos:**
***Engenharia de Software***

**2.1. Study Programme:**
***Software Engineering***

**2.2. Grau:**
***Mestre***

**2.3. Ramo, variante, área de especialização do mestrado ou especialidade do doutoramento (se aplicável):**
***<sem resposta>***

**2.3. Branch, option, specialization area of the master or speciality of the PhD (if applicable):**
***<no answer>***

**2.4. Ano/semestre/trimestre curricular:**
***1º ano / 2º semestre***

**2.4. Curricular year/semester/trimester:**
***1st year / 2nd semester***

**2.5. Plano de Estudos / Study plan**

| Unidade Curricular / Curricular Unit | Área Científica / Scientific Area (1) | Duração / Duration (2) | Horas Trabalho / Working Hours (3) | Horas Contacto / Contact Hours (4) | ECTS | Observações / Observations (5) |
|---|---|---|---|---|---|---|
| Gestão de Projetos, Inovação e Empreendedorismo / Project Management, Innovation and Entrepreneurship | CINF | Semestral | 162 | 14 TP + 28 PL | 6 | Obrigatória |
| Segurança em Engenharia de Software/Security in Software Engineering | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Compreensão e Evolução de Software/Software Evolution and Program Comprehension | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Análise de Dados e Engenharia de Software /Data Analysis in Software Engineering | CINF | Semestral | 162 | 42 TP | 6 | Obrigatória |
| Computação Móvel / Mobile Computing | CINF | Semestral | 162 | 42 TP | 6 | Optativa |
| Gestão de Qualidade e Melhoria de Processos de Software / Quality Management and Software Process Improvement | CINF | Semestral | 162 | 42 TP | 6 | Optativa |
| Qualquer unidade curricular de U.Porto (2º ciclo)*/ Any course of U. Porto (2nd cycle)* | CINF / QACUP | Semestral | 162 | Depende da UC escolhida | 6 | Optativa;* Sujeita a aprovação prévia do Diretor de do ciclo de estudos. |

**(7 Items)**

**Mapa III - - 2º ano**

**2.1. Ciclo de Estudos:**
***Engenharia de Software***

**2.1. Study Programme:**
***Software Engineering***

**2.2. Grau:**
***Mestre***

**2.3. Ramo, variante, área de especialização do mestrado ou especialidade do doutoramento (se aplicável):**
***<sem resposta>***

**2.3. Branch, option, specialization area of the master or speciality of the PhD (if applicable):**
***<no answer>***

**2.4. Ano/semestre/trimestre curricular:**
***2º ano***

**2.4. Curricular year/semester/trimester:**
***2nd year***

**2.5. Plano de Estudos / Study plan**

| Unidade Curricular / Curricular Unit | Área Científica / Scientific Area (1) | Duração / Duration (2) | Horas Trabalho / Working Hours (3) | Horas Contacto / Contact Hours (4) | ECTS | Observações / Observations (5) |
|---|---|---|---|---|---|---|
| Seminários em Engenharia de Software /Software Engineering Seminars | CINF | Semestral (S1) | 162 | 42 TP | 6 | Obrigatória |
| Computação Orientada a Serviços e Nuvem /Cloud and Service Oriented Computing | CINF | Semestral (S1) | 162 | 42 TP | 6 | Optativa |
| Interação Pessoa Computador/ Human Computer Interaction | CINF | Semestral (S1) | 162 | 42 TP | 6 | Optativa |
| Qualquer unidade curricular de U.Porto (2º ciclo)*/ Any course of U. Porto (2nd cycle)* | CINF / QACUP | Semestral (S1) | 162 | Depende da UC escolhida | 6 | Optativa;* Sujeita a aprovação prévia do Diretor de do ciclo de estudos. |
| Dissertação/Dissertation | CINF | Anual | 1296 | 28 OT | 48 | Obrigatória |

**(5 Items)**

## 3. Descrição e fundamentação dos objectivos, sua adequação ao projecto educativo, científico e cultural da instituição, e unidades curriculares

**3.1. Dos objectivos do ciclo de estudos**

**3.1.1. Objectivos gerais definidos para o ciclo de estudos:**
***O objetivo do MESW é promover a excelência da qualificação em aspetos avançados e nucleares da Engenharia de Software (ES), desde os seus fundamentos teóricos até à integração de tecnologias informáticas nas organizações para que os estudantes possam:***
***• Trabalhar eficientemente numa equipa de projeto para desenvolver software (SW) de qualidade;***
***• Detetar conflitos nos requisitos e definir soluções nos limites de custo, tempo, conhecimentos existentes e idiossincrasias organizacionais;***
***• Projetar soluções de ES atendendo a restrições éticas, sociais, legais, tecnológicas e económicas;***
***• Compreender e valorizar a análise de viabilidade, negociação, hábitos de trabalho eficientes, liderança, comunicação e estudo continuado;***
***• Aprender novos modelos, técnicas e tecnologias de desenvolvimento de SW***

***• Analisar os pontos fortes e fracos da tecnologia de desenvolvimento de SW, fomentando a mudança nas organizações;***
***• Liderar o trabalho em várias áreas do desenvolvimento de SW.***

**3.1.1. Generic objectives defined for the study programme:**
***The mains aim of MESW is to promote the excellence in professional qualification in both core and advanced Software Engineering topics, from its theoretical foundations to integrating software technologies inside organisations, enabling graduates to:***
***•efficiently develop quality software within a project team;***
***•detect and manage requirements conflicts, and devise solutions that balance cost, time, knowledge and organisational constraints;***
***•design and create software engineering solutions, considering ethical, social, legal, economical and technological constraints;***
***•understand and give value to feasibility assessment, negotiation, efficient working habits, leadership, communication skills and lifelong learning;***
***•learn new models, techniques and technologies of software development;***
***•promote organisational change by identifying caveats in the software development technology and suggesting improvements;***
***•coordinate software engineering teams throughout the software development process.***

**3.1.2. Objectivos de aprendizagem (conhecimentos, aptidões e competências) a desenvolver pelos estudantes:**
***O MESW define um conjunto de objetivos de aprendizagem descritos e estruturados com base nos Descritores de Dublin e no referencial EUR-ACE:***
***• Conhecer e compreender de forma crítica e aprofundada os princípios, metodologias e técnicas centrais da Engenharia de Software (ES) designadamente as envolvidas em problemas de complexidade e dimensão elevadas;***
***• Ser capaz de conceber, projetar e implementar soluções e produtos de software novos e complexos usando conhecimento de vanguarda na área;***
***• Ser capaz de julgar de modo crítico novas soluções tecnológicas e promover a busca e a aplicação de métodos e soluções inovadoras, apelando à curiosidade, criatividade e rigor;***
***• Ser capaz de comunicar o seu raciocínio de alto teor tecnológico a público diverso de forma oral ou escrita em contextos (inter)nacionais;***
***• Ser capaz de prosseguir a sua formação em ES de forma autónoma e ao longo da vida;***
***• Ser capaz de pesquisar informação de forma crítica para a resolução de problemas de ES complexos.***

**3.1.2. Intended learning outcomes (knowledge, skills and competences) to be developed by the students:**
***The Masters in Software Engineering defines a set of learning outcomes described and structured according to the Dublin Descriptors and the EUR-ACE referential. Graduates should therefore be able to:***
***•have a thorough grasp and deep understanding of the principles, methodologies and techniques at the core of the Software Engineering (SE) area, particularly those that address large and highly complex problems;***
***•design, plan and implement software solutions and new, complex products, using top-notch, state-of-the-art domain knowledge and technologies;***
***•assess and judge new technological solutions and promote the search and application of innovative solutions and methods, fostered by curiosity, creativity and rigour;***
***•address national and international audiences, in written form or orally, exhibiting efficient communication skills about technology;***
***•autonomously engage in lifelong learning in SE;***
***•research data and information that allows solving complex Software Engineering problems.***

**3.1.3. Inserção do ciclo de estudos na estratégia institucional de oferta formativa face à missão da instituição:**
***A Faculdade de Engenharia da Universidade do Porto desenvolve atividade de educação, investigação e inovação de nível internacional. Em conformidade, são resultados desta atividade a criação, transmissão e difusão de conhecimento, a formação de profissionais competentes e éticos, e futuros líderes na área de engenharia e afins, assim como a promoção do bem-estar da sociedade global.***
***O plano estratégico da FEUP-2011-15 contempla vinte eixos de atuação denominados politicas estratégicas, sendo de destacar, destas,***
***• as relacionadas com a qualidade e internacionalização (políticas estratégicas 1 e 2) que visam promover a qualidade global com critérios internacionalmente aceites e promover a internacionalização, particularmente no sentido do alargamento da base de cooperação internacional;***
***• as relacionadas com a formação em engenharia (política estratégica 3 e 4), que focam a liderança nacional e internacional da educação na área, assim como a reforma da oferta formativa, na estrutura, nos conteúdos e nos métodos de ensino e avaliação em engenharia.***
***O ciclo de estudos (MESW) proposto surge do trabalho excelente que tem sido desenvolvido no grupo de Engenharia de Software da Faculdade de Engenharia da Universidade do Porto ligado a instituições de investigação e ensino de prestígio internacional como, por exemplo, o SEI (Software Engineering Institute) da CMU (Carnegie Mellon University), PARC (Palo Alto Research Center), CERN (European Organization for Nuclear Research), USP (Universidade de São Paulo), University of Maryland, University of Naples Federico II, INESC TEC, e***

***empresas/instituições de renome internacional como Microsoft, IBM, Google, Facebook, NASA (National Aeronautics and Space Administration), entre outras.***
***O MESW pretende contribuir para incrementar os ingressos na FEUP para realização de uma formação de 2º ciclo em Engenharia Informática, na área específica de Engenharia de Software. É importante referir aqui que esta oferta formativa visa atingir públicos-alvo no espaço da lusofonia (tanto nas comunidades portuguesas e de luso-descendentes, como também públicos residentes nos países de Comunidade de Países de Língua Portuguesa) mas também no espaço internacional e pretende-se que seja lecionada em Inglês. A oferta contará no futuro com uma parceria, que está a ser negociada neste momento, com a Universidade de São Paulo (USP) e outras em análise (como por exemplo, ESEO - Institute of Science and Technology, França). Outras parcerias serão analisadas a seu tempo.***
***Pretende-se que esta cooperação internacional fomente a circulação de estudantes, docentes e investigadores na procura contínua de currículos atualizados, enriquecimento mútuo e investigação de ponta.***
***Os mestres em Engenharia de Software são necessários para a indústria de software da mesma forma que os MBAs (Mestrados em Administração de Negócios) o são para a gestão pelo que a necessidade do mercado é evidente.***

**3.1.3. Insertion of the study programme in the institutional training offer strategy against the mission of the institution:**
***The Faculty of Engineering at the University of Porto conducts educational activities, world-class research and innovation. Accordingly, the results of this activity are the creation, transmission and dissemination of knowledge, the training of competent and ethical professionals and future leaders in engineering and the like, and the promotion of the welfare of the global society.***

***The FEUP-2011-15 strategic plan includes twenty actuation axes called strategic policies, most notably,***
***• Those related to quality and internationalisation (strategic policies 1 and 2) to promote overall quality through internationally accepted criteria and to promote internationalisation, particularly aimed at extending the international cooperation base;***
***• Those related to engineering education (strategic policies 3 and 4), which focus on national and international leadership in the area of education, as well as the reform of the range of training, structure, contents and methods of teaching and assessment in engineering.***

***The proposed course (MESW) arises from the excellent work that has been done in the Faculty of Engineering of the University of Porto by the Software Engineering research group, linked to research and education institutions of international prestige, e.g. the SEI (Software Engineering Institute) at CMU (Carnegie Mellon University), PARC (Palo Alto Research Center), CERN (European Organisation for Nuclear Research), USP (Universidade de São Paulo), University of Maryland, University of Naples Federico II, INESC TEC and internationally renowned companies or institutions such as Microsoft, IBM, Google, Facebook and NASA (National Aeronautics and Space Administration).***

***The MESW aims at increasing the number of students at FEUP conducting a 2nd cycle training in Computer Science, in the specific area of Software Engineering. It should be noted here that this training expects to reach audiences in the Lusophone countries (both in Portuguese communities and Portuguese descendants, as well as residents in countries of the Portuguese Speaking Countries Community) as well as in the international arena, by having English as its official language. MESW will in future gain from a partnership which is being negotiated at the moment with the University of São Paulo (USP) and others under analysis (e.g. ESEO - Institute of Science and Technology, France). Other partnerships will be analysed in due course.***

***It is intended that this international cooperation promotes the mobility of students, teachers and researchers in the continuing search for updated curricula, mutual enrichment and cutting-edge research.***

***Masters of Science in Software Engineering are required for the software industry in the same way that the MBA (Masters in Business Administration) are for the management so the need of the market is evident.***

**3.2. Adequação ao projeto educativo, científico e cultural da Instituição**

**3.2.1. Projeto educativo, científico e cultural da Instituição:**
***A missão da Faculdade de Engenharia da Universidade do Porto desenvolve-se essencialmente nas áreas da engenharia e afins, tendo como dimensões principais a formação académica, as atividades de investigação, desenvolvimento e inovação em estreita ligação com as formações de segundo e principalmente de terceiro ciclo e, ainda, as atividades da terceira missão da Universidade que incluem a transferência de conhecimento e tecnologia, a prestação de serviços, a oferta de formação contínua, a participação na discussão de políticas nacionais e o envolvimento na vida económica, cultural e social da nossa região e do país. Deve ver-se como parte integral dessas dimensões, na sua complementaridade, a formação cultural, cívica e humanista da Comunidade FEUP, a valorização da envolvente e do património e a preservação da memória da instituição.***
***Com as alterações ao Ensino Superior decididas na União Europeia (processo de Bolonha), passaram a existir 3 graus: Licenciatura (3 anos), Mestrados (2 anos) e Doutoramentos (3 a 4 anos). Para além destes graus, a Faculdade de Engenharia da Universidade do Porto criou também ciclos de estudos que conjugam Licenciatura e o Mestrado, chamados Mestrados integrados (5 anos).***

***Atualmente a FEUP oferece um Mestrado Integrado em Engenharia Informática e Computação (MIEIC) com uma vocação de formação em banda larga em Engenharia Informática, cobrindo diversas outras áreas além da Engenharia de Software. O Mestrado em Engenharia de Software irá fortalecer a oferta formativa da FEUP na área de Engenharia de Software. Em todo o caso, pretende-se que, sempre que possível, exista uma partilha de unidades curriculares do MESW com unidades curriculares optativas do MIEIC na área de Engenharia de Software, por forma a maximizar a eficiência letiva. Dessa forma, o próprio MIEIC sairá beneficiado, por se dar acesso aos estudantes do MIEIC, através da opção de escolha livre, a um leque alargado, atualizado e coerente de unidades curriculares optativas na área de Engenharia de Software , mantendo o selo de 5 anos de formação na FEUP conferido pelo MIEIC.***
***Contabilizando os graduados do ciclo de estudos predecessor do MIEIC (LEIC-Licenciatura em Engenharia Informática e Computação) e os atuais graduados, o total soma, desde 1999, cerca de 900 profissionais que, na sua maioria, se mantêm na região Norte. Contribuem assim, com o exercício da sua profissão em instituições públicas e privadas, para o desenvolvimento da região. Contudo esta oferta formativa não tem sido suficiente para dar resposta às solicitações do mercado.***

**3.2.1. Institution's educational, scientific and cultural project:**

***The mission of the Faculty of Engineering of the University of Porto is mainly in the fields of engineering and those closely related. Its core activities focus at academic certification and training, research, development and innovation (closely related to 2nd and 3rd study cycle degrees) and also the activities described in the third mission of the University. These include transfer of knowledge and technology, provision of services, provision of training, participation in national policy discussions and involvement in the economic, cultural and social life of the region and the country. The cultural, civic and humanistic training of the FEUP community, the appreciation of its surroundings, its heritage and the preservation of the institution's memory should be seen as an integral yet complementary part of these dimensions.***

***With the changes in Higher Education Policies decided by the European Union (Bologna treaty), academic certification was reduced to three degrees: Bachelor (3 years), Master (2 years) and PhD (3-4 years). In addition to these degrees, the Faculty of Engineering of the University of Porto has also created courses that combine undergraduate and Masters, known as the Integrated Masters (5 years).***

***Currently, FEUP offers a Masters in Informatics and Computing Engineering (MIEIC) with a broad training in Computer Science covering several areas besides software engineering. The MSc in Software Engineering at FEUP will strengthen the training available in Software Engineering. In any case, it is intended that, wherever possible, MESW modules be offered to MIEIC as optional modules in the Software Engineering area, in order to maximise academic efficiency. As such, MIEIC will benefit from it, because their students will have a wider, updated and consistent range of optional modules in the Software Engineering area (shared with MESW), keeping MIEIC's five years "seal" of training conferred at FEUP.***

***Taking into account graduates of both MIEIC's predecessor course (formerly called LEIC - Degree in Informatics and Computing Engineering) and current graduates, the total since 1999 is around 900 professionals, the majority of whom remain in northern Portugal, thus contributing to the development of the region, through the exercise of their profession in public and private institutions. Nevertheless, the range of training currently available has not been enough to meet market demand.***

**3.2.2. Demonstração de que os objetivos definidos para o ciclo de estudos são compatíveis com o projeto educativo, científico e cultural da Instituição:**

***O Mestrado em Engenharia de Software (MESW) pretende ser um ciclo de formação de excelência ao nível do 2º ciclo de Bolonha e tem como principal objetivo formar profissionais com sólida formação de base e competências em Engenharia de Software para suprimir as necessidades do mercado, promover a inovação e incrementar a competitividade.***
***O MESW define objetivos de aprendizagem, operacionalizados pelas suas unidades curriculares e verificados através da análise do percurso académico dos estudantes, tendo como referência o perfil necessário para profissionais capazes de, dada uma necessidade, aplicar uma abordagem sistemática, disciplinada e quantificável conducente ao desenvolvimento, operação e manutenção de um sistema de software que satisfaça os requisitos funcionais e os atributos de qualidade.***
***O ciclo de estudos proposto (MESW) pretende contribuir para incrementar os ingressos na FEUP para realização de uma formação de 2º ciclo em Engenharia Informática, na área específica de Engenharia de Software, por candidatos nacionais e internacionais, dando resposta a necessidades de mercado.***
***A estrutura curricular do MESW cumpre os requisitos definidos no despacho Nº GR.02/06/2014 de alteração ao regulamento geral dos segundos ciclos de estudos da Universidade do Porto. De facto, o MESW está configurado como um ciclo de estudos de quatro semestres correspondentes a 120 créditos cumprindo o estipulado no "Artigo 7º - Duração do ciclo de estudos".***
***O MESW tem uma componente curricular de 72 ECTS e uma componente de dissertação de 48 ECTS cumprindo o estipulado no "Artigo 6º - Estrutura de ciclo de estudos". A componente de dissertação será orientada por um professor ou investigador doutorado da U.Porto cumprindo o estipulado no "Artigo 9º - Orientação da dissertação, do trabalho de projeto ou do estágio".***
***Tal como referido no ponto 3.1.2, o MESW tem como objetivo de aprendizagem a desenvolver pelos estudantes um***

***conjunto de competências fundamentais cumprindo o estipulado no nº 4 do "Artigo 3º - Grau de Mestre.***

**3.2.2. Demonstration that the study programme's objectives are compatible with the Institution's educational, scientific and cultural project:**

***The Master in Software Engineering (MESW) aims at providing training of excellence, at the 2nd cycle of Bologna and to prepare students with solid basic training and skills in software engineering, in order to satisfy market needs, promote innovation and increase competitiveness.***

***MESW learning objectives are upheld by its modules and verified through analysis of the academic path taken by its students. Its definition takes as a reference the profile required for professionals, who are capable of applying a systematic, disciplined and quantifiable approach, leading to the development, operation and maintenance of a software system that meets the functional requirements and quality attributes.***

***The proposed course (MESW) aims at increasing the number of students at FEUP taking a 2nd cycle training in Computer Science, in the specific area of software engineering, for domestic and international applicants, while responding to market needs.***

***The curriculum structure of MESW meets the requirements set out in Order No. GR.02/06/2014 amendment to the general regulations for the second cycle of the University of Porto studies. In fact, the MESW is a four semester course, corresponding to 120 ECTS, fulfilling the provisions of "Article 7 - Course duration."***

***The MESW has a curricular component of 72 ECTS and a 48 ECTS thesis component fulfilling the provisions of "Article 6 - course structure". The dissertation component will be guided by a professor or doctoral researcher at the University of Porto, fulfilling the provisions of "Article 9 - dissertation Guidance, project work and internship."***

***As stated in section 3.1.2, the learning objectives of the MESW aere a set of core skills to be acquired by the students, thus fulfilling the provisions of paragraph 4 of "Article 3 - Master Degree."***

**3.3. Unidades Curriculares**

**Mapa IV - Análise de Dados em Engenharia de Software /Data Analysis in Software Engineering**

**3.3.1. Unidade curricular:**

***Análise de Dados em Engenharia de Software /Data Analysis in Software Engineering***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**

***Carlos Manuel Milheiro de Oliveira Pinto Soares (TP-30 horas)***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**

***Luís Filipe Pinto de Almeida Teixeira (TP-12 horas)***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**

***• Saber identificar problemas, em particular no âmbito de atividades de engenharia de software (ex. gestão de projetos, desenvolvimento e manutenção de software, apoio à pesquisa de informação relevante para responder a dúvidas de programação), em que a utilização de técnicas de extração de conhecimento (EC) de dados, ou data mining, permita melhorar os processos***
***• Desenvolver a capacidade de utilizar corretamente uma abordagem de EC para a resolução de problemas nessa área, particularmente em casos em que envolvam grandes quantidades de dados.***
***• Compreender o funcionamento de alguns algoritmos de EC descritivo (k-means, regras de associação) e preditivo (árvores, modelos lineares, naive bayes , redes neuronais, support vector machines e ensembles)***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**

***• Ability to identify problems, particularly in the area of software engineering (e.g. project management, development and maintenance of software, support in the search for relevant information for addressing doubts), in which the use of techniques of knowledge-discovery in databases (KDD), or data mining (DM) may improve results .***
***• Develop the ability to properly utilize a DM approach to address the problem in that area, particularly in cases involving large amounts of data.***
***• Understand the behaviour of certain DM algorithms, both for descriptive tasks (k-means, association rules) and predictive tasks (trees, linear models, Naïve Bayes, neural networks, support vector machines and ensemble methods).***

**3.3.5. Conteúdos programáticos:**

***•Introdução à EC (Data Mining), incluindo exemplos ilustrativos em Engenharia de Sofware.***

***Data Mining descritivo***
***•Clustering: Algoritmos de partição, hierárquicos e outros. Medidas de avaliação.***
***•Regras de associação: Algoritmo APRIORI. Outros algoritmos. Medidas de avaliação.***
***•Metodologias de Data Mining: Processo de EC. Metodologia CRISP-DM. Gestão de projetos.***
***•Pré-processamento de dados: Limpeza de dados e transformação de dados.***
***Data Mining preditivo***
***•Avaliação de modelos preditivos: Árvores de decisão. Sobre-ajustamento em árvores de decisão. Metodologias de avaliação.***
***•Classificação: Algoritmos de classificação (baseados em regras, distâncias e kernels; métodos Bayesianos). Questões comuns em classificação (distribuição de classes desequilibrada e custos). Medidas de avaliação.***
***•Regressão: Algoritmos de regressão (regressão linear e não linear, árvores de regressão, MARS). Medidas de avaliação.***
***Análise de Dados complexo***
***• Text mining: Representação de dados para text mining.***

**3.3.5. Syllabus:**
***•Introduction to knowledge extraction/data mining, including illustrative examples in Software Engineering.***
***Descriptive of DM***
***•Clustering: Partitional, hierarchical and other algorithms. Evaluation measures.***
***•Association Rules: A priori algorithm and other algorithms. Evaluation measures.***
***•Methodologies for DM: The process of knowledge extraction. CRISP-DM. Project management.***
***•Pre-processing of data: Data cleansing and transformation.***
***Predictive DM***
***•Evaluation of predictive models: Decision trees. Overfitting in decision trees. Evaluation methodologies.***
***•Classification: Classification algorithms (rule-, instance- and kernel-based methods, Bayesian methods). Common Issues in classification (unbalanced distribution of classes and costs). Evaluation measures.***
***•Regression: Regression algorithms (linear and non-linear regression, regression trees, MARS). Evaluation measures.***
***Analysis of Complex data***
***Text mining: Representation of data for text mining. Evaluation measures.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***A matéria apresentada nas aulas bem como os exercícios e o projeto vai permitir aos estudantes aprender a identificar problemas em que a utilização de técnicas de extração de conhecimento (EC) de dados, ou data mining, permita melhorar os processos de engenharia de software e também outros domínios de aplicação.***

***A matéria apresentada vai também permitir aos estudantes compreender o funcionamento de um conjunto diversificado de algoritmos.***

***Finalmente, os exercícios em aula e, em especial, o projeto, vai-lhes permitir desenvolver a capacidade de utilizar corretamente uma abordagem de EC para a resolução de problemas.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***The material presented in class, as well as the exercises and the project will enable students to learn to identify problems in the use of data mining techniques, and will improve the software engineering process and also in applications in other fields of application.***

***The material presented will also enable students to understand how a diverse set of algorithms works.***

***Finally, class exercises and the project in particular will allow students to develop the ability to properly use a DM approach to solving problems***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***Metodologias de ensino***
***Aulas teórico-práticas, incluindo***
***• exposição e discussão de conceitos.***
***• Exercícios para aplicação prática dos conceitos aprendidos.***
***Avaliação***
***Avaliação distribuída com exame final***
***(0.50* Projeto + 0.5 * Nota do exame)***

**3.3.7. Teaching methodologies (including assessment):**
***Teaching methods***

***Practical classes, including***
***• Presentation and discussion of the concepts.***
***• Exercises for practical application of the concepts learned.***
***Assessment***
***Continual assessment with final exam***
***(0.5* Assignment Grade + 0.5* Exam Grade)***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***As aulas servem para apresentar e discutir os conceitos essenciais de Extração de Conhecimento (EC) e ilustrar esses conceitos com exemplos de projetos de Engenharia de Software em que os docentes tenham participado ou que estejam descritos na literatura. O exame serve essencialmente para aferir se esses conceitos essenciais foram adquiridos pelos estudantes.***

***A componente mais prática da unidade curricular é extremamente importante porque algumas das questões essenciais para o sucesso dos projetos de EC só podem ser ilustradas no âmbito de aplicações reais e/ou de exercícios práticos.***

***Assim, as sessões mais práticas servem para os estudantes adquirirem prática na utilização de ferramentas para a resolução de problemas de EC. Os problemas usados nestas sessões são preparados de forma a focarem nas questões que se pretende avaliar e, ao mesmo tempo, serem realizáveis no tempo disponível. Sempre que possível, serão usados problemas da área de Engenharia de Software.***

***Por outro lado, o projeto complementa os exercícios das aulas, expondo os estudantes a um contexto menos estruturado de desenvolvimento de projetos de EC, no âmbito da Engenharia de Software. Este tipo de contexto cria desafios em várias fases do processo de EC, incluindo na definição do problema, preparação dos dados (em particular na sua limpeza), avaliação dos resultados e sua otimização, e na gestão dos projetos. Duas outras formas de aproximar os projetos da realidade de uma aplicação de EC são: os estudantes são encorajados a recolherem dados para os respetivos projetos, em vez dos benchmarks tipicamente usados pela comunidade de EC na sua investigação; os docentes fazem o papel de cliente, em particular na definição do problema, na compreensão dos dados e na avaliação dos resultados.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***The lectures serve to present and discuss the essential concepts of Knowledge Extraction (KDD) and illustrate these concepts with examples of projects from the field of Software Engineering that the teachers have participated in or that are described in the literature. The examination is essentially to ascertain whether these essential concepts were acquired by the students.***

***The practical part of the course is extremely important because some of the key issues for the success of KDD projects can only be illustrated in the context of real applications and/or practical exercises.***

***Therefore, the more practical sessions are for students to gain experience in the use of software tools to solve KDD problems. The problems used in these sessions are prepared to focus on the specific issues that are the goal of the corresponding exercise and, at the same time, be solvable in the available time. When possible, problems from the area of Software Engineering will be used.***

***Additionally, the project complements the exercises in the classes , by exposing students to less structured KDD development projects, which are from the Software Engineering field. This type of environment creates challenges in many of the phases of the KDD process , including problem definition, data preparation (particularly in data cleaning), evaluation of results and optimisation, and management of projects. Two other ways to approaching these projects from the reality of a KDD application are: students are encouraged to collect data for the respective projects, instead of the benchmarks that are typically used by the KDD community in their research; teachers take on the role of the customer, particularly in the definition of the problem, understanding of the data and analysis of the results.***

**3.3.9. Bibliografia principal:**

***Halkidi, M., Spinellis, D., Tsatsaronis, G., & Vazirgiannis, M. (2011). Data mining in software engineering. Intelligent Data Analysis, 15(3), 413–441.***
***Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann Publishers.***
***Maimon, O., & Rokach, L. (2010). Data Mining and Knowledge Discovery Handbook. Vasa. doi:10.1007/978-0-387-09823-4***
***Mendonça, M., & Sunderhaft, N.L., (1999). Mining software engineering data: A survey. A DACS state-of-the-art report, Data & Analysis Center for Software, Rome, NY (1999).***
***Xie, T., Thummalapenta, S., Lo, D., & Liu, C. (2009). Data Mining for Software Engineering. Computer, 42(8), 55–62. doi:10.1109/MC.2009.256***

**Mapa IV - Arquitetura e Desenho de Software/Software Architecture and Design**

**3.3.1. Unidade curricular:**
***Arquitetura e Desenho de Software/Software Architecture and Design***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Ademar Manuel Teixeira de Aguiar; TP-30 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Hugo José Sereno Lopes Ferreira; TP-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***A arquitetura d um sistema de software descreve a sua estrutura global em termos dos componentes,das propriedades externas desses componentes e das suas interrelações.Para sistemas d média e grande dimensão a escolha adequada da arquitetura assume uma importância crucial para o sucesso do seu desenvolvimento.A UC tem como objetivo principal dotar os estudantes com os conceitos essenciais de arquiteturas de software,padrões d desenho e tópicos diretamente relacionados,bem como o de componentes de software.Pretende que os estudantes fiquem habilitados a desenhar,compreender e avaliar arquiteturas d sistemas d software,tanto ao nível d abstração d macro-arquitetura como de micro-arquitetura e assim familiarizar os estudantes com os conceitos fundamentais de arquitetura de software,as propriedades e aplicabilidade dos diferentes estilos existentes,os padrões de desenho mais populares,componentes,arquiteturas reutilizáveis e as relações destes conceitos todos com a reutilização de software.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***The architecture of a software system describes the global structure in terms of its components, external properties and its interrelations. For medium and large-size systems, choosing the right architecture is crucial to the success of its development.***
***This module aims to provide students with the concepts of software architecture, design patterns and directly related topics such as software components. It also aims to enable students to design, understand and evaluate software system architecture both at the level of abstraction of macro-architecture and micro-architecture. All this is to familiarize the students with the fundamental concepts of software architecture, its proprieties and the applicability of the different architecture styles, the most popular design patterns, components, reusable architectures and the relationship of all these concepts with software reuse.***

**3.3.5. Conteúdos programáticos:**
***Introdução. Desenho de software: conceitos, princípios e abordagens fundamentais. O que é arquitetura de software? A importância da arquitetura de software. Exemplos de arquiteturas aos níveis micro e macro: padrões de desenho, frameworks e linhas de produção. Estilos de Arquitetura. Os principais estilos arquiteturais. Exemplos clássicos Padrões de Desenho: As origens dos padrões de software. Tipos de padrões de software: padrões de arquitetura, padrões de desenho, estruturas idiomáticas. Exemplos de Padrões de Arquitetura: POSA patterns. Padrões de Desenho: GoF patterns. Arquiteturas de Software: Arquiteturas e Atributos de Qualidade. Estilos de arquiteturas, modelos de referência e arquiteturas de referência. Desenho, avaliação e refinamento de arquiteturas de software. Representação e Documentação de arquiteturas de software. Reutilização de arquiteturas de software: linhas de produção, frameworks, componentes de software.***

**3.3.5. Syllabus:**
***Introduction. Software Design: fundamental concepts and principles. What is software architecture? The importance of software architecture. Examples of micro and macro architecture: design patterns, frameworks and production lines. Architecture Styles Design Patterns: Origin of software patterns. Types of software patterns: architecture patterns, design patterns, idiomatic structures. Examples of Architecture Patterns: POSA patterns. Examples of Design Patterns: GoF patterns. Software Architecture: Quality Architecture and Attributes. Architecture styles, reference models and architecture. Design, evaluation and refinement of software architecture. Representation and Documentation of software architecture. Reuse of software architecture: production lines, frameworks, software components.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Os conteúdos abordados na UC visam educar os estudantes para a sistematização do processo de definição de arquiteturas de sistemas de software, quer a alto-nível quer ao nível de desenho detalhado, desde o levantamento***

***dos requisitos funcionais e não funcionais, reutilização de macro e micro arquiteturas na forma de padrões de software, até à sua concretização e avaliação critica. Para atingir estes objetivos é importante conhecer os principais estilos de arquitetura, padrões de arquitetura, padrões de desenho, bem como saber representar e documentar as arquiteturas.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The contents of the module aim at teaching students a systematic process for defining the architecture of software systems, both high-level and low-level design, from the functional and non-functional requirements, reuse of macro and micro architecture as software patterns, to its realisation and critical analysis. To attain these objectives, it is important to know the key architectural styles, architecture patterns, design patterns, as well as to represent and document architectures.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***As aulas teóricas são utilizadas tanto para a exposição formal dos principais conhecimentos da unidade curricular como para a apresentação final e discussão dos projetos desenvolvidos pelos estudantes ao longo do semestre. Por forma a focar a atenção dos estudantes nos tópicos principais, serão propostas pequenas questões de resposta facultativa e livre sobre os tópicos em estudo para desenvolvimento fora de aulas. Na componente prática da UC, os estudantes terão oportunidade de colocar em prática os conhecimentos transmitidos através de pequenos exercícios, jogos de simulação e desenvolvimento incremental de um projeto de média dimensão. As questões, os exercícios e o projeto conjuntamente incentivarão os estudantes a complementar os conhecimentos transmitidos com outros conhecimentos resultantes de pesquisas individuais efetuadas sobre os conteúdos da unidade curricular.***

***Fórmula de avaliação: 30% exame escrito, 30% trabalho escrito, 30% trabalho prático, 10% avaliação de desempenho.***

**3.3.7. Teaching methodologies (including assessment):**

***Lectures – formal presentation of contents and final presentation and discussion of the projects undertaken by the students during the semester. In order to focus students' attention on the main topics, small tests with multiple-choice answers will be given to be, to be done out of class. During the semester, students will have the chance to put the knowledge acquired into practice, by doing short exercises, simulation games and the incremental development of a medium-sized project. The questions, exercises and project will encourage the students to complement the knowledge acquired with other know-how resulting from the individual research done on the module contents.***

***Assessment: 30% written exam, 30% written assignments, 30% project, 10% individual evaluation.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***As aulas são coerentes com os objetivos da unidade curricular. A transmissão formal de conhecimentos nas aulas, a realização de exercícios e trabalhos sobre tópicos avançados e relevantes, bem como a discussão de casos de estudo nas aulas, irão conjuntamente permitir aos estudantes ganharem um conhecimento mais aprofundado sobre os tópicos da UC, aprender a aplicá-los, bem como adquirirem capacidade critica. Adicionalmente, os estudantes realizarão trabalhos práticos em grupo, durante e após as aulas, por forma a consolidar a aprendizagem e dar experiência prática.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***The classes are consistent with the objectives of the course. The formal transmission of knowledge in the classroom, the resolution of exercises and assignments on relevant advanced topics, as well as the discussion of case studies, will all combine to help students acquire a more solid knowledge of the various topics, learn how to apply that knowledge in practice, and to gain critical analysis skills on the topic. Students will also undertake practical work in groups, during and after classes, in order to consolidate what they have learnt and also to provide them with practical experience.***

**3.3.9. Bibliografia principal:**

***Bass, L., Clements, P., & Kazman, R. (2003). Software architecture in practice (2nd ed.).***

***Gamma, E. (1995). Design patterns: Elements of reusable object-oriented software. Reading, Mass: Addison-Wesley.***

***Buschmann, F.., Meunier R.., Rohnet H.., Sommerland P.., & Stal M.. (1996). Pattern-Oriented Software Architecture.***

***Clements, P., Bachmann, F., Bass, L., Garlan, D., Ivers, J., Little, R., Nord, R.,, Stafford, J. (2011). Documenting Software Architectures: Views and Beyond. Boston, MA: Addison-Wesley.***

**Mapa IV - Compreensão e Evolução de Software/Software Evolution and Program Comprehension**

**3.3.1. Unidade curricular:**

***Compreensão e Evolução de Software/Software Evolution and Program Comprehension***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Rui Filipe Lima Maranhão de Abreu; TP-30 horas.***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Nuno Honório Rodrigues Flores; TP-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***Depois de concluir esta unidade curricular, o estudante será capaz de***
***- Gerir a evolução e manutenção de software***
***- Lidar com mudança de software***
***- Aplicar processos de manutenção de software***
***- Aplicar engenharia reversa de software***
***- Identificar antipadrões e padrões de software***
***- Lidar com evolução em linhas de produto de software***
***- Gerir a reengenharia de software***
***- Utilizar ferramentas de manutenção***
***- Evolução e desenvolvimento de código aberto***
***- Identificar desafios importantes de lista associados com evolução de software;***
***- Discutir métodos e ferramentas para abordar esses desafios, as suas vantagens edesvantagens;***
***- Aplicar os métodos e ferramentas para sistemas de software existente;***
***- interpretar os resultados obtidos de forma cientificamente responsável***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***After completing this course, you will be expected to be able to deal with***
***- software maintenance and evolution***
***- software change***
***- software maintenance processes***
***- software reverse engineering***
***- software patterns and anti-patterns***
***- software product lines***
***- software reengineering***
***- maintenance tools***
***- open source development***
***- listing important challenges associated with software evolution;***
***- discussing methods and tools addressing these challenges, their advantages and disadvantages;***
***- applying the methods and tools to existing software systems;***
***- interpreting the results obtained in a scientifically responsible way.***

**3.3.5. Conteúdos programáticos:**
***Evolução e manutenção de software***

***Identificação e remoção de software clones***

***Análise de Repositórios de Software para compreender a evolução do Software***
***Prevenir erros com base no passado do desenvolvimento***

***Reengenharia orientada a objetos***

***Migração do legado de sistemas de informação***

***Transformações arquitectorais***

***A interação entre testes de Software e evolução e seu efeito na compreensão do programa***

***Questões de evolução em programação orientada a aspectos***

***Evolução de arquitetura de software***

***Estudos empíricos de evolução de código aberto***

**3.3.5. Syllabus:**
***Software maintenance and evolution***

***Identifying and removing software clones***
***Analysing Software Repositories to understand software evolution***
***Predicting bugs from history***
***Object-oriented reengineering***
***Migration of legacy information systems***
***Architectural transformations***
***The interplay between software testing and evolution and its effect on program comprehension***
***Evolution issues in aspect-oriented programming***
***Software architecture evolution***
***Empirical studies of open source evolution***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***Os conteúdos programáticos foram definidos em função dos objetivos e competências a serem adquiridos pelos estudantes. Os conteúdos programáticos incluem as abordagens consideradas estado-da-arte em compreensão e evolução de software bem como técnicas específicas que conferem aos estudantes a capacidade de seleccionar de forma crítica o método apropiado a utilizar para resolver problemas de engenharia. Ao mesmo tempo, os estudantes serão capazes de interpretar e analisar o comportamento dos algoritmos e os resultados obtidos pelos mesmos.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The syllabus was defined according to the objectives and competencies to be acquired by the students. The syllabus includes understanding state-of-the-art approaches to software comprehension and evolution as well as specific techniques which will make students capable of critically selecting the appropriate method to use to solve the engineering problem at hand. At the same time, students will be able to interpret and analyse the behaviour and performance of the algorithms and the results obtained by them.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***ENSINO***
***As aulas terão uma componente de exposição e discussão dos tópicos programáticos, e outra componente de realização de exercicíos práticos e pequenos projetos de aplicação dos conceitos e técnicas e experimentação de ferramentas. Será reservada uma ou duas aulas para a apresentação de tópicos explorados pelos estudantes.***

***AVALIAÇÃO***
***1) 1 exame final (individual; 30% da nota final)***
***2) 10 trabalhos de analíse critica de técnicas abordadas (grupo de 2 estudantes; 1 trabalho por semana; 30% da nota final)***
***3) Projeto (40%; grupo de 4 estudantes de forma a criar a complexidade ao que o número de membros numa equipa diz respeito)***

**3.3.7. Teaching methodologies (including assessment):**

***TEACHING***
***The classes will comprise the presentation and discussion of topics and the development of practical exercises and small projects by the students. Slots will be reserved for the presentation of special topics explored by the students (having an article and presentation as output).***

***ASSESSMENT***
***1) 1 final exam (individual; 30% of final grade)***
***2) 10 assignments of critical analysis of state-of-the-art techniques (Groups of 2 students; 1 assignment per week; 30% of final grade)***
***3) project (40%; groups of 4 students in order to create the complexity generated by the number of members in a team)***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***O desenvolvimento das aulas decorrerá harmonizando as metodologias de ensino com os objetivos fundamentais da unidade curricular (UC). Esta será uma UC de aplicação, onde os estudantes aprenderão não só a teoria, mas também como executar, avaliar e decidir.***
***O fornecimento de informação e de conhecimentos científicos e técnicos previstos nos objetivos será desenvolvido nas aulas teóricas, onde será estabelecida a relação com outras matérias já tratadas em aulas anteriores ou noutras UCs. Nestas sessões pretender-se-á desenvolver as competências dos estudantes e sensibilizá-los para a importância dos temas abordados no contexto cientifico-empresarial, contribuindo-se para um melhor enquadramento e também maior facilidade na perceção dos objetivos que se pretendem alcançar.***

***Dado o carácter eminentemente prático das matérias a versar, serão apresentados e propostos nas aulas teórico-***

***práticas vários exercícios e casos práticos obtidos do contacto com empresas parceiras. Os estudantes aprenderão fazendo, refletindo e tomando decisões sobre os problemas e alternativas propostas, melhorando as suas competências nos temas em análise.***

***Tentar-se-á estimular um processo de diálogo em que todos participem, através da sua própria experiência e saber. Assim, partilhar-se-á conhecimento, dúvidas e questões, de modo a beneficiar a aprendizagem dos estudantes e a provocar maior motivação dos mesmos. Procurar-se-á, essencialmente, garantir o desenvolvimento das capacidades de "aplicar em contextos diferentes" os conhecimentos adquiridos, sob influência de diferentes fatores e variáveis.***
***O trabalho prático de grupo exigido aos estudantes terá um importante contributo para a realização dos objetivos definidos para a UC, proporcionando a compreensão e a aplicação das temáticas em estudo. Este trabalho permitirá identificar os diferentes recursos e componentes de um projeto, suas relações internas e externas, bem como utilizar de forma geral e integrada os conceitos e metodologias abordados ao longo desta e de outras unidades curriculares.***
***A realização do trabalho prático terá ainda as vantagens de partilha de conhecimentos entre os elementos do grupo, procura de informação externa e, portanto, contacto com a realidade. A sua posterior apresentação e defesa, bem como a análise de um projeto realizado por outro grupo da turma, contribuirão de modo decisivo para o reforço da capacidade de análise que se considera essencial para a consecução dos objetivos desta UC.***

***A avaliação dos estudantes servirá para a aferição da eficácia das metodologias de ensino desenvolvidas na observância dos objetivos da UC e, se necessário, no futuro poder-se-á realizar algumas correções nas metodologias de ensino.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**
***The intention is that the teaching methodologies used in classes will suit the fundamental objectives of the module. This will be an applied-oriented module, in which students will learn not only the theory, but also how to execute, evaluate and take decisions.***
***The scientific and technical knowledge that the students will acquire during this module, will be taught in lectures, and also by establishing a relationship with previously acquired knowledge and with other modules. In these lectures, the aim is to develop the students' skills and make them aware of the importance of the topics covered, from scientific and business points of view, and contributing to a framework and a better understanding of the aims to be achieved.***
***Given the eminently practical nature of the topics covered by this module, exercises and case studies obtained from partner companies will be discussed and studied in the practical classes. Students will learn by doing, reflecting and making decisions about the problems and potential alternatives, improving their skills in the topics addressed.***
***This module will try to stimulate a dialogue amongst all students, bringing to bear their own experience and knowledge. Sharing knowledge, doubts and questions in this way will benefit student learning and lead to greater motivation. It will essentially motivate the capacity to "apply in different contexts" the knowledge acquired under the influence of different factors. The group assignment will have an important contribution for the students to properly acquire the required knowledge for this module, enabling them to comprehend and apply the topics studied. This assignment will help students to identify different aspects of a software project, its internal and external relationships, as well as the general and integrated usage of the concepts and methodologies learnt in this and other modules.***
***The practical assignment brings also the advantage of sharing knowledge amongst the members of the group, seek external information, and therefore emulating a living lab for the real world. The final presentation, as well as critical discussion of a project presented by another group in the class, will contribute to the capacity of the students to properly defend and assess complex projects.***
***The assessment of the students will serve to understand the efficacy of the teaching methodologies used in the module to achieve its aims, and, if necessary, adjustments can be made in the future.***

**3.3.9. Bibliografia principal:**
***Software Evolution, Mens and Demeyer, Springer, 2008***

***Software Fundamentals, edited by Daniel M. Hoffman and David M. Weiss, Addison Wesley, 2001***
***Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Addison-Wesley Professional,1994***
***Refactoring: Improving the Design of Existing Code, Addison-Wesley Professional, 1999***

**Mapa IV - Computação Móvel / Mobile Computing**

**3.3.1. Unidade curricular:**
***Computação Móvel / Mobile Computing***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**

***António Miguel Pontes Pimenta Monteiro TP: 42 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Não aplicável/Not applicable***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***Os objetivos de aprendizagem visam a aquisição de conhecimento e a prática dos conceitos do projeto, arquitetura e implementação de aplicações, principalmente empresariais, em dispositivos móveis, utilizando plataformas e APIs de grande divulgação corrente. São problemas e desafios a vencer a grande diversidade de capacidades, formas de interação e componentes pouco usuais dos dispositivos móveis atuais. Fazem também parte dos objetivos o conhecimento e a prática do desenvolvimento de aplicações de razoável dimensão em diversos ambientes e a integração de serviços remotos.***
***Os estudantes, dada uma especificação de requisitos, deverão ser capazes de:***
***- escolher uma plataforma móvel, tecnologias, padrão da aplicação, 'framework' de programação, e a adequação às necessidades de armazenamento e comunicação;***
***- desenvolver e realizar aplicações e serviços locais e remotos, nas principais tecnologias;***
***- integrar serviços e aplicações disponíveis, mesmo de diferentes tecnologias.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***The learning objectives include the acquisition of knowledge and practice of the concepts of design,architecture and implementation of applications, mainly for business, on mobile devices,using the wide current range of platforms and APIs.There are problems and challenges to meet due to the wide range of capabilities,forms of interaction and unusual components amongst today's mobile devices.***
***Important requirements include a good knowledge and practice in the development of local and remote components in various environments.The development,use, and integration of remote services(web services)is also very useful.***
***Presented with a requirements brief, students should be able to:***
***-choose a mobile platform,technolgies,application architecture pattern,programming framework,and communication and storage needs suitable for the needs;***
***-implement the local application and UI and needed local and remote services;***
***-integrate available applications and services,even from different technologies.***

**3.3.5. Conteúdos programáticos:**
***Conceitos teóricos:***
***1.Aplicações para dispositivos móveis***
***-características adaptadas aos dispositivos móveis***
***-necessidades e facilidades de comunicação***
***-armazenamento local***
***2.Padrões para aplicações nos dispositivos móveis***
***-aplicações locais***
***-aplicações ligadas e desligadas***
***-partilha da lógica de negócio***
***-necessidade de dados locais e remotos***
***-caches***
***-bases de dados locais e sincronização***
***3.Utilização de 'frameworks' de programação e desenvolvimento correntes***
***-Android, Windows Phone e iOS***
***-aspectos da construção de interfaces e navegação***
***-utilização de sensores***
***-comunicações***
***-acesso a dados locais e remotos***
***4.Aplicações orientadas aos serviços***
***-acesso a web services***
***5.Distribuição e instalação de aplicações nos dispositivos***
***6.Alguns aspectos de segurança***
***Prática:***
***Desenvolvimento de aplicações para dispositivos móveis utilizando 'frameworks' correntes e genéricos. Utilização de ferramentas de desenvolvimento e emuladores dos dispositivos.***

**3.3.5. Syllabus:**
***Theory:***
***1. Applications and interfaces for mobile devices***
***- Mobile devices characteristics***
***- Communications facilities and needs***
***- Local storage***

***2. Patterns for applications on mobile devices***
***- Local applications***
***- Disconnected and connected applications***
***- sharing business logic***
***- Local and remote data needs***
***- Caching***
***- Local data bases and synchronization***
***3. Use of current and generic programming frameworks***
***- Android, Windows Phone, iOS***
***- User interface design, programming and navigation***
***- Sensors***
***- Communications***
***- Local and remote data access***
***4. Service-oriented applications***
***- Use of web services***
***5. Installation and deployment of applictions in the mobile devices***
***6. Security***
***Practice:***
***Development of mobile applications using current and generic programming frameworks.***
***Development tools and the use of mobile device emulators.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***Nesta unidade curricular começa-se por caracterizar as aplicações para dispositivos móveis, as suas necessidades e especificidades assim como os padrões habituais utilizados no seu desenvolvimento. Esses padrões minimizam as dificuldades decorrentes da grande diversidade das plataformas móveis. Introduzem-se alguns 'frameworks' de utilização comum, assim como o desenvolvimento, para aplicações empresariais, de serviços externos em sistemas remotos, num ambiente distribuído. São ainda abordados aspetos de instalação e segurança e a utilização de componentes de software e hardware pouco usual.***
***A transmissão desta informação faz-se não só por via da exposição, mas também através de numerosos exemplos, demonstrações e trabalhos desenvolvidos pelos estudantes.***
***Essa prática inclui o projeto e o desenvolvimento de pelo menos duas aplicações de grande dimensão em ambientes de programação diferente (Java, .NET, …) envolvendo normalmente serviços e dados remotos (como REST web services e outros).***
***Todos estes tópicos e respetiva prática contribuem decisivamente para permitir os objetivos enunciados de um sólido conhecimento e a prática da competente conceção e desenvolvimento de software que explore a componente móvel e a sua integração em sistemas distribuídos empresariais e comunicação remota.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***In this module we begin by the characterisation of mobile applications, their needs and specificities, as well as the usual patterns used in their development. These patterns minimise the difficulties caused by the great diversity of mobile platforms. Next, some concrete frameworks of great use are described and practiced. The development and use of external services in remote systems, essential for enterprise applications, is also included. Some aspects of deployment, security and the use of unusual software and hardware components are also addressed.***
***This knowledge is not only taught in lectures but especially by various examples, demonstations and work undertaken by the students.***
***This practice includes the design and development of at least two large-scale applications in different programming environments (Java, .NET, ...) usually involving services and remote data (such as REST web services).***
***These topics and their practice contribute decisively to fulfilling the stated objectives of solid knowledge and competent practice in the conception and implementation of software that explores mobile components and their integration in distributed business and remote communication systems.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***São utilizados dois aspetos importantes da transmissão de conhecimento, sendo todas as aulas divididas entre esses dois aspetos:***
***- Exposição teórica dos conceitos e apresentação de exemplos e demonstrações;***
***- Conceção e desenvolvimento prático de diversos tipos de aplicações para dispositivos móveis.***

***Tipo de Avaliação: Avaliação distribuída com exame final***
***Condições de Frequência: É obrigatória a realização e apresentação dos trabalhos.***
***Exame final com consulta.***
***Fórmula de avaliação: 0.6 Tr + 0.4 Ex***
***Tr - 2 a 3 trabalhos práticos a realizar durante o semestre***
***Ex - Exame final com consulta.***

**3.3.7. Teaching methodologies (including assessment):**

***Two important teaching methods are used to transmit knowledge, and all classes are divided between the two:***

***- Lectures, presenting examples and demonstrations of concepts.***
***- Design and practical development of various types of application for mobile devices.***

***Assessment: Continuous assessment, plus final exam***
***Attendance conditions: Completing and presenting the Lab Assignments is mandatory.***
***Open book final.***
***Assessment weighting: 0.6 Coursework + 0.4 Ex***
***Coursework - 2 to 3 Lab assignments***
***Exam - Open book final***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***Esta unidade curricular inclui aulas de índole teórico-prática (3h/semana), que são divididas numa parte de caráter mais expositivo (1h) e noutra essencialmente de projeto e desenvolvimento (2h).***
***A primeira parte contempla a explicação de conceitos, padrões ou métodos de abordagem, sempre com a apresentação de exemplos e demonstrações, seguida de uma parte de aplicação com a resolução de pequenos problemas que incluem o desenvolvimento e execução de pequenas aplicações (ou partes) que exercitem e demonstrem alguns dos conceitos apresentados.***
***A unidade curricular inclui também pelo menos dois trabalhos práticos de grande envergadura pedindo-se a conceção, implementação e demonstração de um sistema completo, centrado numa aplicação móvel, mas incluindo também serviços ou comunicações externas. Este trabalho é em geral a solução para um cenário empresarial significativo.***
***Os trabalhos procuram sempre abranger mais do que uma tecnologia, nomeadamente requerendo-se que, em trabalhos diferentes, pelo menos o uso de dois frameworks móveis distintos. As demonstrações poderão ser feitas utilizando os emuladores geralmente disponibilizados pelos sistemas de desenvolvimento, ou dispositivos reais, se os estudantes os possuírem.***
***Todas estas ações e métodos de aprendizagem, baseados sobretudo em pequenos projetos e em fazer concretamente, mas sem esquecer a aprendizagem de conceitos e metodologias de aplicação geral, contribuem de forma decisiva para o atingir dos objetivos propostos e a aquisição das competências necessárias à exploração deste tipo de sistemas***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***This module only involves practical classes (3h/week), but they are always divided into a first more explanatory part (1h) and a second part essentially for design and implementation (2h), taking place in a lab.***
***The first part involves the presentation of concepts and methodology, according to the syllabus, and the second part is for application, where some examples and demonstrations are presented and some small problems are solved, including the development and execution of small applications (or parts of bigger ones), exercising and demonstrating the concepts presented.***
***The module also includes at least two assignments of broader scope, requiring the design, implementation, deployment and demonstration of a complete system, focused on a mobile application, but also including remote services and communications. These assignments frequently involve the solution of some significant business scenario problem.***
***The assignments always seek to include more than one technology and its integration, and it is required that, in different assignments, at least two different mobile frameworks should be used. The demonstrations can be done using the emulators usually available in the development systems, or real devices, if the students own them.***
***All these actions and learning methods are based mainly on projects and actually doing, without neglecting the learning of general concepts, patterns and methods in common use, which contribute decisively to attaining the proposed objectives and the skill and practice acquisition needed for the exploitation of this kind of system.***

**3.3.9. Bibliografia principal:**

***- Dino Esposito, "Architecting Mobile Solutions for the Enterprise", Microsoft Press, 2012, ISBN: 978-0-735-66302-2***
***- Adrian Mendoza, "Mobile User Experience - Patterns to Make Sense of it All", Morgan Kaufmann, 2014, ISBN: 978-0-12-409514-4***
***- Daniel Sauter, "Rapid Android Development", The Pragmatic Programmers, 2013, ISBN: 978-1-93778-506-2***
***- Dave MacLean , Satya Komatineni , Grant Allen, "Pro Android", 5th edition, Apress, 2015, ISBN: 978-1-4302-4680-0***
***- Wallace Jackson, "Pro Android Wearables", Apress, 2015, ISBN: 978-1-4302-6550-4***
***- Andrew Whitechapel, Sean McKenna, "Windows Phone 8 Development Internals", Microsoft Press, 2013, ISBN: 978-0-7356-7623-7***
***- James Bucanek, "Learn iOS 8 App Development", 2nd edition, Apress, 2014, ISBN: 978-1-484202-09-8***

**Mapa IV - Computação Orientada a Serviços e Nuvem /Cloud and Service Oriented Computing**

**3.3.1. Unidade curricular:**

***Computação Orientada a Serviços e Nuvem /Cloud and Service Oriented Computing***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Hugo José Sereno Lopes Ferreira (TP-30 horas)***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***António Miguel Pontes Pimenta Monteiro (TP-12 horas)***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***Conhecimentos de Engenharia de Software, nomeadamente técnicas e processos de computação baseado em arquitecturas orientadas a serviços, com especial foco em ambientes sobre virtualização das camadas infraestruturais (vulgarmente conhecidos por "nuvem" ou cloud). Pretende-se desenvolver as capacidades de gerir todo o ciclo de vida de software, desde a sua análise, desenho e concepção, até à implementação e manutenção, compreendendo as especificidades nos algoritmos, boas práticas e padrões utilizados neste domínio. No final da unidade curricular os estudantes devem ser capazes de desenhar e implementar software de larga escala em ambientes falíveis, sujeitos a picos elevados de stress, baixa latência, alta largura de banda e performance, compostos por um elevado número de componentes heterogéneos sujeitos a interferências e interacções imprevistas.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***Knowledge of Software Engineering, namely techniques and processes of computation based on service-oriented architectures, with a special focus on environments with virtualized infrastructures (commonly known as clouds). The goal is to develop the capabilities to tackle the full software development lifecyle, from analysis, design and conception, through to the implementation, maintenance and deployment of a domain with very specific algorithmic needs, good practices and patterns. By the end, students should be able to design and implement large scale software systems, highly resilient to failure, subject to bursts of stress, low latency, high performance and bandwidth, composed of a large number of interacting heterogeneous components with non-trivial interferences.***

**3.3.5. Conteúdos programáticos:**
***Fundamentos: Arquitecturas clássicas versus baseadas em serviços (SOAs). Micro-serviços. Standards e padrões de comunicação de serviços. Webservices e SOAP versus REST. Barramentos. Princípios e modelos de computação paralela e distribuída. Modelo de actores. Modelos e arquitecturas map-reduce/lambda. Limites teóricos e práticos. Virtualização de Infraestruturas. PaaS, SaaS e IaaS. Clouds públicas e Privadas. Armazenamento. Elasticidade. High-Availability e replicação. Clusters. Monitorização e Diagnóstico. Falhas e tolerância a falhas. Latência e Interferência. Sistemas reactivos e Streams. Segurança. Aplicações: Paradigma de programação objecto-funcional com Scala. Modelo de actores com Akka. Arquitecturas MapReduce com Hadoop e Spark. Clouds públicas com Amazon AWS e Google Computing Engine. Clouds privadas com software livre (OpenStack). Bases de dados key-value. HAProxy. Barramentos baseados em AMQP. ZeroMQ. Técnicas de deployment contínuo, zero de tempos mortos.***

**3.3.5. Syllabus:**
***Fundamentals: Classic vs service-oriented architectures. Microservices. Communication protocols and standards. Webservices, SOAP versus REST. Data buses. Principles and models of parallel and distributed computing. Actor model. Map-reduce/lambda arquitectures and models. Practical and fundamental limits. Infrastructure virtualization. PaaS, SaaS and IaaS. Public and private clouds. Storage. Elasticity. High-Availability and replication. Clusters. Reactive systems and Streams. Security. Applications: Object-Functional Programming Paradigm with Scala. Actor model with Akka. MapReduce architectures with Hadoop and Spark. Public clouds with Amazon AWS and Google Computing Engine. Private clouds with open-source based software (OpenStack). Key-value databases. HAProxy. Data buses based on AMQP. ZeroMQ. Continuous deployment and zero down-time techniques.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***O conteúdo programático segue uma estrutura que vai acompanhando o ciclo de vida de software, bem como o nível de abstracção entre o software e a respectiva infraestrutura. Começando no desenho, as arquitecturas clássicas e baseadas em (micro-serviços) capacitam o estudante para compreender padrões e estilos arquitectónicos largamente utilizados e de que forma estes tentam dar solução às necessidades deste domínio. Seguem-se formas de intercomunicação entre os componentes, numa primeira fase a espelhar as ténicas clássicas e comunicação inter-processos, mas rapidamente seguindo um percurso diferenciador. Pretende-se igualmente capacitar de noções e principios matemáticos fundamentais que constrigem fortemente o espaço de possíveis soluções, tornando assim o acto de "desenho" mais rigoroso, recorrendo-se para tal a modelos cujas abstracções estão bem definidas e estudadas (e.g., modelo de atores e map-reduce). Continuando a percorrer os diferentes níveis, passamos agora a compreender o conceito de virtualização das infra-estruturas, principalmente as computacionais, de rede e de armazenamento, que permitem o suporte instantânea de sistemas de larga escala. Passamos a entender a forma como as economias de escala são afectadas e novos serviços são criados, aumentando consideravelmente uma cadeia de valor que antes estava centrada no conceito físico de "máquina": Paas, SaaS e IaaS. Dada a total virtualização da infraestrutura, entramos finalmente em pleno conceito de "nuvem", e potenciamos o aparecimento de novas técnicas entrega do software (eg. entrega contínua e sem tempos mortos)***

***e novas formas de escalamento (eg. escalamento elástico) com impacto económico directo. Surgem também toda uma nova classe de necessidades com impacto profundo na última etapa do diclo de vida de software: a manutenção. São finalmente abordadas novas considerações de segurança, bem como o aumento do impacto e responsabilidade social da engenharia informática. Em paralelo, todos estes conceitos são acompanhados de case-studies industriais que permitem uma reflexão mais aprofundada dos conhecimentos, bem como de ferramentas e tecnologias que permitem aos estudantes colocá-los em prática imediata.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The syllabus structure very closely follows the software lifecycle, as well as the abstraction level between the software and the corresponding infrastructure. Starting at the design level, the study of classic and micro-service based architectures empower the student to understand widely used architectural patterns and styles and how they attempt to provide solutions for this domain. We then proceed to study inter-component communication, by following closely at first classic techniques of inter-process communication, but quickly moving into differentiation factors. Students will likewise be presented with mathematical principles that constrain the solution space, making the act of "design" more rigorous by relying on models in which abstractions are well-defined and studied (e.g. actor model and map-reduce). Progressing to the next levels, the students now dive into the concept of infrastructure virtualization, mainly computational, network and storage, which allows the instantaneous support of large-scale systems. They now start to understand how new scale economies and new services were created that added considerably to a value chain once focused on the physical concept of a "machine", namely by looking at PaaS, SaaS and IaaS. Given the full infrastructure virtualization, we now go fully into the concept of "cloud", along with new deployment (continuous delivery, zero downtime), and scaling techniques (elasticity) with a profound economic impact. It should now be evident that all this has a direct impact on the last phase of the software lifecycle: maintenance. Finally, new security considerations are studied in the light of the increased social impact and responsibility from the field. In tandem, all these concepts are accompanied by industrial case studies that allow a deeper reflection on the knowledge, as well as the usage, of tools and technologies and allow its immediate application.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***A UC tem uma componente teórica inicial de exposição e discussão, que incluem a descrição dos problemas, as metodologias de análise e as soluções e boas práticas preconizadas. Segue-se de uma componente prática dedicada à aplicação e aprofundamento dos conceitos e técnicas discutidas através de exercícios, pequenos projetos e trabalhos de pesquisa. As componentes de avaliação são (i) projeto semi-guiado, onde os estudantes irão ao longo da UC proceder a todo o ciclo de desenvolvimento de um software orientado a serviços e entregue na nuvem, mas pontualmente guiados através de pequenos tutoriais que visam introduzir os conceitos necessários à sua conclusão, com um peso de 40% (ii) trabalho de pesquisa, que permite aos estudantes focarem-se numa área de interesse e aprofundarem conhecimentos mais específicos, fazendo uma breve exposição final para avaliação, com um peso de 20% e (iii) exame final de aferição de conhecimentos teóricos, com um peso de 40%.***

**3.3.7. Teaching methodologies (including assessment):**

***The course has an initial theoretical component based on recitations and discussion that include the description of the problems, analysis methodologies, and their solution and known good practices. It is followed by a more practical approach dedicated to the concrete application and deepening of the discussed concepts and techniques, through exercises, small projects and research work. The assessment is thus (i) semi-guided project, where the students undertake the development of the full lifecycle of a service-oriented software system on the cloud, albeit occasionally guided in tutorials that incrementally present new and required concepts, weighted at 40% (ii) research work, that will enable students to focus their learning on specific topics, with a brief final presentation, weighted at 20%, and (iii) a final, more theoretical, exam, weighted at 40%.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***As aulas são coerentes com os objetivos da unidade curricular. Em particular, a transmissão de conhecimentos nas aulas, a realização de exercícios e discussão de casos de estudo nas aulas, o aprofundar e apresentar um trabalho prático aos restantes colegas e a realização de um trabalho autónomo de pesquisa, permitem estabelecer uma plataforma que sustenta a aprendizagem dos estudantes.***

***Em particular, a compreensão das especificidades deste domínio e o seu impacto no ciclo de vida de desenvolvimento de software necessita de uma forte componente prática, onde os estudantes vivenciam as dificuldades e ideossincrasias do desenho baseado em serviços e nuvem. Desta forma, o trabalho prático vai permitir aos estudantes interagir com as várias ferramentas actualmente disponíveis para desenhar um sistema real e executar esse sistema num ambiente em nuvem. Este trabalho é apoiado pelo docente, mas os estudantes devem procurar chegar a soluções autonomamente. Devem apresentar um relatório final com as vários etapas efetuados e apresentar publicamente o resultado numa aula. Igualmente, o forte leque que se abre de novas técnicas, possiblidade e preocupações que ultrapassam o âmbito de numa única unidade curricular, torna importante o aprofundamento individual em tópicos mais especializados, objectivo para o qual contribui o trabalho de pesquisa, que exigirá um nivel de autonomia mais elevado que o trabalho prático, e que será igualmente defendido na aula final. Por fim, a avaliação individual dos conhecimentos mais teóricos é complementada através***

***do exame escrito.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**
***Classes are consistent with the objectives of the module. In particular, the transmission of knowledge in the classroom, conducting exercises and discussion of case studies in the classroom, the deepening and presentation of a developed software to the other colleagues and performing an independent research work, establish a platform that supports student learning.***

***In particular, the understandement of the specificities of this domain and its impact on the software development lifecycle requires a strong practical component, where the students directly feel the difficulties and idiosyncrasies of service-oriented and cloud-based design. Hence, developing a cloud software will allow the students to interact with the various tools abailable to draw a real system and deploy it in a cloud environment. This work is supported by the teacher, but students should try to reach independent solutions. They should present a final report with the various steps required to reach that particular solution and present it during a specific class. Due to the large number of possibilities and techniques that are now available but don't fit into a single module, it is important for the student to search for more specific topics of interest through the research work, which will require a more autonomous learning effort and will also be presented in the final class. Finally, the individual assessment of more theoretical concepts is complemented by the written exam.***

**3.3.9. Bibliografia principal:**
***Hohpe and Woolfe, "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions", Addison Wesley, 2003.***
***Van Roy et al. "Concepts, Techniques, and Models of Computer Programming", MIT Press 2004.***
***Rosen et al. "Applied SOA: Service-Oriented Architecture and Design Strategies", Wiley 2008.***
***George Reese. "Cloud Application Architecture", O'Reilly 2009.***
***Odersky and Spoon. "Programming in Scala 2e", Artima 2011.***
***Kuhn and Allen. "Reactive Design Patterns", Manning 2015.***
***Roestenburg et al. "Akka in Action", Manning 2015.***

**Mapa IV - Dissertação/Dissertation**

**3.3.1. Unidade curricular:**
***Dissertação/Dissertation***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Raul Fernando de Almeida Moreira Vidal - OT-28 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**

***Docentes deste ciclo de estudo e a definir posteriormente de acordo com o tema da Dissertação.***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***Trabalho individual de investigação e desenvolvimento, conducente à elaboração de uma dissertação de natureza científica sobre um tema da área de conhecimento do ciclo de estudos. Pode ser um trabalho de investigação ou de desenvolvimento tecnológico e aplicação, envolvendo meios experimentais e/ou de simulação, que promova o desenvolvimento de capacidades de iniciativa, de decisão, de inovação, de pensamento criativo e crítico, num contexto de trabalho individual ou em grupo. Deve envolver a análise de situações novas, a recolha de informação pertinente, o desenvolvimento e selecção ou concepção das metodologias de abordagem e dos instrumentos de resolução do problema proposto, a sua resolução, o exercício de síntese e elaboração de conclusões, e a preparação de uma dissertação pertinente sujeita a apresentação pública dos resultados.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***Individual Work, research and development leading to a scientific dissertation on a topic within the area of the course. It may be a work of research or technological development and implementation, involving experimental media and/or simulation, which promotes the development of capacity for initiative, decision-making, innovation, creative and critical thinking, individually or within a group. It should involve the analysis of new situations, the collection of relevant information, development and selection of design or approach methodologies and instruments proposed for resolving problems, the synthesis and drawing of conclusions, and the preparation of a relevant dissertation subject to public presentation and discussion of results.***

**3.3.5. Conteúdos programáticos:**
***Não há matéria curricular específica, pois os estudantes da Dissertação irão debruçar-se sobre múltiplos temas,***

***cobrindo os mais variados domínios do conhecimento respeitantes ao ciclo de estudos e, pela integração e aplicação desses conhecimentos e das competências e atitudes antes adquiridos, irão dedicar-se à resolução de problemas complexos de engenharia de software.***

**3.3.5. Syllabus:**

***There is no specific syllabus, because dissertation students will investigate a number of topics covering various fields of knowledge relating to the course and, by the integration and application of knowledge, skills and attitudes acquired before, will devote themselves to the resolution of complex engineering software problems.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***Como já foi dito não existem conteúdos programáticos específicos, dada a natureza da unidade curricular. Os temas abordados pelos estudantes e os respectivos trabalhos desenvolvidos serão, naturalmente, escolhidos de acordo com os objectivos de aplicação de conhecimentos, competências e atitudes adquiridos ao longo do ciclo de estudos.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***As noted above, there is no specific syllabus given the nature of the module. The topics covered by students in their projects will naturally be selected in accordance with the main goal of applying the knowledge, skills and attitudes acquired during the course.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***Trabalho individual de investigação e desenvolvimento, podendo ser realizado em ambiente académico ou académico/empresarial. Neste caso os objetivos, natureza e forma de acompanhamento do trabalho devem ser objeto de acordo prévio e de permanente monitorização através de reuniões entre o estudante, o orientador por parte da FEUP e o responsável pelo acompanhamento dos trabalhos na empresa.***

**3.3.7. Teaching methodologies (including assessment):**

***Individual research and development work that can be undertaken in an academic or academic/business environment. The objectives, nature and form of monitoring will be subject to prior agreement, and there will be constant monitoring through meetings between the student, the supervisor from FEUP and the person responsible for monitoring the work in the company.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***Nesta Unidade Curricular espera-se que os estudantes desenvolvam o trabalho de forma mais autónoma deixando-lhes a maior responsabilidade de trabalharem o seu tema de forma apropriada e consentânea com os objetivos da UC. Pretende-se que os estudantes mostrem que sabem aplicar os conhecimentos, competência e atitudes adquiridos ao longo do ciclo de estudos. Os orientadores acompanharão o trabalho por forma a que se cumpra o plano de trabalhos proposto.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***Students are expected in this module to undertake their work indepedently, taking most of the responsibility to work on their topics in accordance with the aims of the module. They are expected to demonstrate that they are able to apply the knowledge, skills and attitudes they acquired during the course. Supervisors will monitor student's work in such a way that guarantees that the work plan is being followed.***

**3.3.9. Bibliografia principal:**

***Martha Davis; "Scientific Papers and Presentations". ISBN: 0-12-206370-8, Committee on Science, Engineering, and Public Policy,; "On Being a Scientist: Responsible Conduct in Research file", National Academies Press, 2009. ISBN: ISBN: 0-309-11971-5,***
***R. S. WAZLAWICK; "Metodologia de Pesquisa para Ciência da Computação", Elsevier, 2009. ISBN: 978-85-352-3522,***
***Justin Zobel; "Writing for computer science". ISBN: 978-1-85233-802-2.***
***Luís Adriano Oliveira; "Dissertação e Tese em Ciência e Tecnologia Segundo Bolonha", Lidel, 2011. ISBN:***
***IEEE Computer Society "Guide to the Software Engineering Body of Knowledge" (SWEBOK) V. 3.0 ISBN: 13:978-0-7695-5166-1***
***Mais bibliografia poderá ser sugerida de acordo com o tema de cada trabalho.***
***More bibliography will be suggested depending on the topics covered by the theme of each work.***

**Mapa IV - Engenharia de Requisitos e Modelação de Software/ Requirements Engineering and Software Modeling**

**3.3.1. Unidade curricular:**
***Engenharia de Requisitos e Modelação de Software/ Requirements Engineering and Software Modeling***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Ana Cristina Ramada Paiva; TP-30 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Gil Manuel Magalhães de Andrade Gonçalves; TP-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***O objetivo desta unidade curricular é ensinar os métodos, técnicas e melhores práticas para planear, gerir e realizar um processo de análise e especificação de requisitos de um sistema de tecnologias baseadas em serviços.***

***No final do ciclo de estudos, os estudantes deverão ter as habilidades necessárias para planear, gerir e realizar um processo de análise e especificação de requisitos de um sistema de software e/ou baseado em tecnologias.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***The objective of this module is to teach the methods, techniques and best practices to plan, manage and carry out a process analysis and requirements specification of a system of services-based technologies.***

***By the end of the module, students should have the skills to plan, manage and carry out a process of analysis and requirements specification of a software and/or technology-based system.***

**3.3.5. Conteúdos programáticos:**
***1. Introdução: âmbito e importância da engenharia de requisitos (ER); noção de requisito; tipos de requisitos; classificação de requisitos; processo de ER; contexto do ciclo de vida dos sistemas.***
***2. Levantamento de Requisitos: fontes de requisitos; técnicas de levantamente de requisitos.***
***3. Análise e negociação de requisitos: fundamentos da negociação de requisitos, gestão de conflitos, técnicas de negociação.***
***4. Especificação de requisitos: documentação em linguagem natural, estruturação de documentos de requisitos.***
***5. Validação de requisitos: fundamentos da validação de requisitos, técnicas de validação.***
***6. Gestão de requisitos: gestão de requisitos, gestão de alterações de requisitos, rastreabilidade de requisitos, ferramentas de gestão de requisitos, prioritização de requisitos.***

**3.3.5. Syllabus:**
***1. Introduction: scope and importance of requirements engineering (RE); notion of requirement; types of requirements; classification of requirements; process of RE; systems life-cycle context.***
***2. Requirements discovery: source of requirements; elicitation techniques.***
***3. Analysis and negotiation of requirements: fundamentals of requirements negotiation, conflict management, negotiation techniques.***
***4. Requirements specification: natural language documentation, structuring requirements documents.***
***5. Requirements validation: fundamentals of reqirements validation, validation techniques.***
***6. Requirements management: requirements management, change management for requirements, requirements traceability, requirements management tools, prioritizing requirements.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Os conteúdos abordados na UC visam educar os estudantes para seguirem um processo engenharia de requisitos que lhes permita fazer um levantamento de requisitos com qualidade e, com isso, contribuir para o desenvolvimento de sistemas/serviços com mais qualidade. Para tal é fundamental (i) conhecer as técnicas fundamentais de suporte ao levantamento de requisitos; (ii) conhecer as técnicas de negociação para obter requisitos aceites por todos os stakeholders, e (iii) conhecer os modelos e standards para documentar requisitos; (iv) saber proceder à validação dos requisitos e gerir todo o processo de engenharia de requisitos; (v) realização de um projeto de grupo com clientes reais para aplicar os conceitos e técnicas estudadas.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***The subjects covered by the module aim at educating students to follow a requirements engineering process that enables them to make a requirements elicitation with quality and, therefore, contribute to the development of systems/services with better quality. For this it is essential (i) to know the fundamental techniques to support requirements gathering, (ii) to know the techniques of negotiation for requirements accepted by all stakeholders, (iii) to know the models and standards for documenting requirements, (iv ) to know the techniques for validating the requirements and managing the entire process of requirements engineering (v) to complete a group project with real customers to apply the concepts and techniques studied.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***As aulas serão com base na apresentação e discussão de temas da unidade curricular, bem como sobre a execução de alguns exercícios com o apoio de ferramentas e na execução de uma parte dos trabalhos práticos nas últimas aulas do semestre. Os estudantes, em grupos de dois, tem que realizar um trabalho prático sobre a especificação de requisitos de um sistema de serviços baseados em tecnologias.***
***Tipo de Avaliação: Avaliação distribuída com exame final***
***Condições de Frequência: A nota mínima de 45% na componente de avaliação contínua.***

***Fórmula de avaliação: A componente de avaliação contínua (50% da nota final) compreende:***
***- Participação nas aulas (5%);***
***- Trabalhos práticos (45%) (Trabalho de modelação e especificação de requisitos: 25%; Palestra sobre um tema de RE: 20%)***
***- Exame Final (50% da nota final)***
***Para aprovação é necessário um mínimo de 45% no exame final.***

**3.3.7. Teaching methodologies (including assessment):**

***Classes will be based on the presentation and discussion of module topics, as well as on the performance of some exercises with the support of tools and on the execution of a part of the practical assignments in the last classes of the semester. Students, working in groups of two, have to carry out a practical assignment on the specification of requirements for a service system based on technologies.***

***Assessment: Continuous assessment plus final exam***
***Attendance requirements: A minimum grade of 45% in the continuous assessment component.***

***Assessment weighting: Continuous assessment grade (50% of the final grade) comprises:***
***- Participation in the classes (5%);***
***- Practical work (45%): (Modeling and requirements specification: 25%; lecture on a topic of RE: 20%);***
***- Final Exam (50% of the final grade).***
***For approval, a minimum grade of 45% in the final exam is required.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***As aulas são coerentes com os objetivos da unidade curricular. Em particular, a transmissão de conhecimentos nas aulas, a realização de exercícios e discussão de casos de estudo nas aulas, o aprofundar e apresentar um tema de engenharia de requisitos aos restantes colegas e a realização de um trabalho de engenharia de requisitos com clientes reais permitem estabelecer uma plataforma que sustenta a aprendizagem dos estudantes.***
***Em particular, o trabalho de levantamento de requisitos com clientes reais permite que os estudantes exercitem o planeamento de reuniões, selecionem técnicas a usar no levantamento de requisitos, definam a forma de documentar requisitos e, por fim, validem os requisitos. Este trabalho de campo é apoiado pelo docente, mas os estudantes devem procurar chegar a soluções autonomamente. Devem apresentar um relatório final com levantamento de requisitos efetuado e apresentar publicamente o resultado na aula final.***
***Assim, para além das metodologias de ensino serem coerentes com os objetivos definidos para a aprendizagem, estimula-se também o desenvolvimento de outras competências sociais aplicáveis em muitos domínios ("soft skills") relevantes para um profissional na área de engenharia de requisitos.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***Classes are consistent with the objectives of the module. In particular, the transmission of knowledge in the classroom, conducting exercises and discussion of case studies in the classroom, the deepening and presentation of requirements engineering topics to the other colleagues and performing a requirements engineering assignment with real clients can establish a platform that supports student learning.***
***In particular, the survey work with real clients' requirements allows students to conduct planning meetings, select techniques to use in gathering requirements, defining how to document requirements and finally validate the requirements. This work is supported by the teacher, but students should try to reach solutions independently. Students must submit a final report with requirements elicitation and present the result publicly in the final class. Thus, in addition to the teaching methodologies being consistent with the goals set for learning, they also stimulate the development of other social skills applicable in many domains ("soft skills") relevant to a professional in the area of requirements engineering.***

**3.3.9. Bibliografia principal:**

***Klaus Pohl; Requirements Engineering: Fundamentals, Principles, and Techniques, Springer; 1st Edition. edition (July 23, 2010), 2010. ISBN: 978-3-642-12577-5***
***Karl E. Wiegers; Software Requirements, 3rd Edition, Microsoft Press, 2013. ISBN: 978-0735618794***
***Howard Podeswa; UML for the it business analyst. 2009, ISBN: 978-1-59863-868-4***

**Mapa IV - Gestão de Projetos, Inovação e Empreendedorismo /Project Management, Innovation and Entrepreneurship**

**3.3.1. Unidade curricular:**
***Gestão de Projetos, Inovação e Empreendedorismo /Project Management, Innovation and Entrepreneurship***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Gil Manuel Magalhães de Andrade Gonçalves (14 horas TP; 4 horas PL)***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Rui Filipe Lima Maranhão de Abreu (8 horas PL)***
***Raul Fernando de Almeida Moreira Vidal (8 horas PL)***
***Ademar Manuel Teixeira de Aguiar (8 horas PL)***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***Desenvolver nos estudantes as capacidades de gestão de projetos de software, trabalho em equipa, inovação e empreendedorismo, habilitando-os assim a resolver a diversidade de problemas que tipicamente surgem no desenvolvimento de um produto tecnológico ou no lançamento de uma empresa de base tecnológica.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***To develop in students software project management, team work, innovation and entrepreneurial skills in the context of software projects. Students will be capable of solving a vast diversity of problems, which may happen in the development of a technology-based product or the launch of a technology-based company.***

**3.3.5. Conteúdos programáticos:**
***Gestão de Projetos: corpo de conhecimento e áreas de conhecimento; metodologias estruturados (RUP, TSP) e agéis (XP, SCRUM); ciclo de vida e fases de um projeto; ferramentas e normas***
***Inovação: gestão de inovação;ciclo de inovação; gestão de projetos de IDI; ferramentas e normas***
***Empreendedorismo: lean startup; modelos de negócio e desenvolvimento de mercado; desenvolvimento e validação de ideias de negócio; produto mínimo viável***

**3.3.5. Syllabus:**
***Project management: body of knowledge and areas of knowledge; structured (RUP, TSP) and agile methodologies (XP, SCRUM); life cycle and phases of a project; tools and standards***
***Innovation: Innovation management; innovation cycle; management of Innovation, research and development projects; tools and standards***
***Entrepreneurship: lean startup; business models and market development; development and validation of business models; minimum viable product***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Esta unidade curricular aborda sobre diferentes abordagens a gestão de projetos de software, práticas de gestão de inovação e modelos de empreendedorismo aplicáveis em empresas de base tecnológica.***
***Embora exista nesta unidade curricular uma componente de exposição, de exemplos e casos, a transmissão de conhecimento processa-se maioritariamente através da execução de projetos por equipas de estudantes.***
***A exposição destes tópicos e respetiva prática contribuem decisivamente para permitir aos estudantes familializar-se com o corpo de conhecimento relacionado com gestão de projetos, inovação e empreendedorismo.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***This module addresses different approaches to software project management, innovation management practices and entrepreneurial models applicable to technology-based companies. Although this module also uses presentations, examples and cases studies, knowledge transmission is mainly driven by practice, by teams of students developing software projects.***
***The presentation of these topics and their practice will contribute decisively to enable students to familiarise themselves with the body of knowledge related to project management, innovation and entrepreneurship.***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***A UC é baseada no trabalho em equipas de média dimensão(4-5 elementos).Aulas TP são usadas para apresentar e discutir tópicos do programa.Haverá também palestras convidadas,apresentações pelas equipas e trabalho fora da sala de aula.As aulas laboratoriais (PL) serão usadas para desenvolvimento de um projeto.Avaliação distribuída sem exame final.Condições de Frequência:Não exceder o número limite de faltas e participar ativamente no planeamento e concretização dos projetos.Relativamente às equipas, a classificação final será calculada com base na participação e discussão nas aulas e o trabalho desenvolvido,tendo em atenção:***
***-Comunicação, imagem e marketing;***

***-Gestão do projeto;***
***-Presença e participação nas aulas e reuniões;***
***-Produto desenvolvido;***
***-Modelo d negócio e exploração dos resultados do projeto.***
***Projetos são avaliados nas componentes d gestão d projeto e trabalho d equipa (40%), inovação (25%), empreendedorismo (25%), presença e participação nas aulas e reuniões (10%).***

**3.3.7. Teaching methodologies (including assessment):**
***The module is based on the work done in medium-sized teams (4-5 members).***
***Practical classes are used to present and discuss syllabus topics. There will also be guest lecturers, presentations by the teams and work outside the classroom.***
***Practical classes in the laboratory will be used to undertake a project.***
***Continuous assessment without final exam.***
***Attendance requirements: Do not exceed the absence limit and actively participate in the planning and implementation of projects.***
***With regard to the teams, the final classification will be calculated based on participation and discussion in class and the work carried out, taking into account:***
***-Communication,image and marketing;***
***-Project management;***
***-Attendance and participation in classes and meetings;***
***-Product developed;***
***-Business model and exploitation of project results.***
***Projects are assessed on the following components: management (40%), innovation (25%), entrepreneurship (25%), attendance and participation in class and meetings (10%).***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**
***Esta unidade curricular será um laboratório vivo, possibilitando aos estudantes uma experiência de aprendizagem na primeira pessoa do que é na realidade desenvolver um produto tecnológico ou criar uma empresa de base tecnológica. O resultado final expetável não é uma apresentaçao ou um relatório, mas um produto e o corresponde modelo de negócio. Os estudantes vão ter de contactar clientes, parceiros e concorrentes, e vão encontrar a complexidade e incerteza inerentes ao verdadeiro funcionamento de uma startup. Os estudantes vão trabalhar em equipa e aprender como transformar boas ideias em grandes produtos ou grandes empresas. Os estudantes vão aprender a utilizar um modelo de negócio e como validar; vão aprender como gerir a inovação e integrar inovação em novos produtos ou empresa; vão aprender a planear e gerir projetos e a desenvolver produtos.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**
***This module will be a living lab, providing real world, hands-on learning experience on what it is like to actually develop a technology-based product or company. The expected end result is not a power-point presentation or a report, but an actual product and the corresponding business model. Students will have to talk to customers, partners, competitors and will encounter the complexity and uncertainty of how a start up really works. Students will work in teams and learn how to turn good ideas into great products or great companies. Students will learn how to use a business model and how to validate it; how to manage innovation and how to bring innovation into their products or companies; how to plan and manage projects and develop products.***

**3.3.9. Bibliografia principal:**
***DeMarco, T. Lister, T. (1999). Peopleware, ISBN: 0932633439.***
***McConnell, S. (1997). Software Project Survival Guide: How to Be Sure Your First Important Project Isn't Your Last, Microsoft Press, ISBN: 1572316217.***
***Jones, C. (2010). Software Engineering Best Practices, McGraw-Hill, ISBN: 9780071621618.***
***Humphrey, W. (2006). TSP: Leading a Development Team, Addison-Wesley, ISBN: 0321349628.***
***IEEE (2011). 1490-2011 IEEE Guide - Adoption of PMI Standard A Guide to the Project Management Body of Knowledge, IEEE Press, E-ISBN: 9780738168173.***
***Humphrey, W. (2005). PSP: A Self-Improvement Process for Software Engineers, Addison-Wesley, ISBN: 0321305493.***
***Osterwalder, A., Pigneur, Y. (2010). Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers, Wiley. ISBN: 978-0470876411.***

**Mapa IV - Gestão de Qualidade e Melhoria de Processos de Software/QualityManagement and Sof.ProcessImprovement**

**3.3.1. Unidade curricular:**
***Gestão de Qualidade e Melhoria de Processos de Software/QualityManagement and Sof.ProcessImprovement***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**

***João Carlos Pascoal de Faria; TP-30 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Raul Fernando de Almeida Moreira Vidal;TP- 12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***No final desta unidade curricular pretende-se que os estudantes:***
***1. Reconheçam a importância da qualidade de produtos, processos e serviços de software e da sua gestão;***
***2. Conheçam as principais atividades e indicadores na gestão da qualidade de produtos, processos e serviços de software;***
***3. Conheçam e saibam implementar modelos de boas práticas de gestão e melhoria do processo de desenvolvimento de software nas organizações, como CMMI-DEV;***
***4. Conheçam modelos de boas práticas de gestão e melhoria de serviços nas organizações, como CMMI-SVC e ITIL;***
***5. Conheçam e saibam implementar técnicas quantitativas de gestão da qualidade e melhoria de processos ao nível pessoal e da equipa, como as incluídas no Personal Software Process/Team Software Process;***
***6. Conheçam e saibam implementar técnicas Lean Six Sigma para a gestão da qualidade do software.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***By the end of this module, it is intended that students:***
***1. Recognise the importance of quality of software products, processes and services and of their management;***
***2. Know the main activities and indicators for quality management of software products, processes and services;***
***3. Know and be able to implement models of best practices for managing and improving the software development process in organizations, such as CMMI-DEV;***
***4. Know models of best practices for service management in organisations, such as CMMI-SVC and ITIL;***
***5. Know and be able to implement quantitative techniques for quality management and process improvement at the personal and team level, such as the ones included in the Personal/Team Software Software Process;***
***6. Known and be able to implement Lean Six Sigma techniques for software quality management.***

**3.3.5. Conteúdos programáticos:**
***1. Estado atual e importância da qualidade de produtos, processos e serviços de software e da sua gestão;***
***2. Principais atividades e indicadores na gestão da qualidade de produtos, processos e serviços de software;***
***3. Modelos de boas práticas de gestão e melhoria do processo de desenvolvimento de software nas organizações, como CMMI-DEV, e metodologias de avaliação e implementação associadas;***
***4. Modelos de boas práticas de gestão e melhoria de serviços nas organizações, como CMMI-SVC e ITIL, e metodologias de avaliação e implementação associadas;***
***5. Técnicas quantitativas de gestão da qualidade e melhoria de processos ao nível pessoal e de equipa, como as incluídas no Personal Software Process/Team Software Process, e metodologias de implementação associadas;***
***6. Aplicação de técnicas de Lean Six Sigma para a gestão da qualidade de software.***

**3.3.5. Syllabus:**
***1. Current status and importance of quality of software products, processes and software and of their management;***
***2. Main activities and indicators for quality management of software products, processes and services;***
***3. Models of best practice for managing and improving the software development process in organisations, such as CMMI-DEV, and related assessment and implementation methodologies;***
***4. Models of best practice for service management in organisations, such as CMMI-SVC and ITIL, and related assessment and implementation methodologies;***
***5. Quantitative techniques for quality management and process improvement at the personal and team level, such as those included in the Personal/Team Software Process, and related implementation methodologies;***
***6. Application of Lean Six Sigma techniques for software quality management.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Os tópicos programáticos estão mapeados, um para um, com os objetivos de aprendizagem.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***The items in the syllabus are mapped one to one with the learning outcomes.***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***ENSINO***
***As aulas terão duas componentes principais: (i) componente teórica de exposição e discussão dos tópicos programáticos e de casos de estudo; (ii) componente prática de realização de pequenos exercícios, bem como de análise de casos práticos e elaboração de recomendações em relação a organizações concretas (de preferência***

***organizações de onde os estudantes são oriundos).***

***AVALIAÇÃO***
***1) Exame final individual (peso 50%);***
***2) Trabalho prático de análise de um caso prático e elaboração de recomendações para uma organização concreta (peso 35%);***
***3) Trabalho de pesquisa e exposição de um tópico relacionado com a unidade curricular (peso 15%)***

**3.3.7. Teaching methodologies (including assessment):**
***TEACHING***
***The classes will comprise two components: (i) presentation and discussion of topics and case studies; (ii) practical exercises, and analysis of practical cases and preparation of recommendations for concrete organisations by the students.***

***ASSESSMENT***
***1) Final exam (individual; 50% of the final grade);***
***2) Practical work of analysis of a pratical case and preparation of recommendations for a concrete organisation (35% of the final grade);***
***3) Researching and presenting a topic related to the module syllabus (15% of the final grade).***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**
***A exposição teórica e discussão de tópicos programáticos e casos de estudo permitirá aos estudantes conhecer os principais modelos de referência, processos e técnicas para a gestão da qualidade e melhoria contínua de produtos, processos e serviços de baseados em software, atingindo assim parte dos objetivos de aprendizagem. Adicionalmente, a realização de exercícios e trabalhos práticos dotará os estudantes da capacidade de implementar algumas das técnicas e modelos de referência estudados.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**
***The theoretical presentation and discussion of module topics and case studies will enable students to be acquainted with the main reference models, processes and techniques for the quality management and continuous improvement of software-based products, processes and services, thus achieving part of the learning objectives. The practical work will also provide students with the ability to implement in practice some of the techniques and reference models studied, thus achieving the remaining learning objectives.***

**3.3.9. Bibliografia principal:**
***Sommerville, I. (2011). Software engineering (9th ed.). Addison-Wesley.***
***Chrissis, M. B., Konrad, M., Shrum, S. (2011). CMMI for Development: Guidelines for Process Integration and Product Improvement (3rd ed.). Addison-Wesley Professional.***
***Forrester, E., Buteau, B., Shrum, S. (2011). CMMI for Services: Guidelines for Superior Service (2nd ed.). Addison-Wesley Professional.***
***Jones, E. (2014). Quality Management for Organizations Using Lean Six Sigma Techniques. CRC Press.***
***Jones, C. (2010). Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies. McGraw-Hill.***
***Humphrey, W., Over, J. (2011). Leadership, Teamwork, and Trust: Building a Competitive Software Capability. Addison-Wesley Professional.***
***Humphrey, W. (2005). PSP: A Self-Improvement Process for Software Engineers. Addison-Wesley Professional.***

**Mapa IV - Interação Pessoa Computador/ Human Computer Interaction**

**3.3.1. Unidade curricular:**
***Interação Pessoa Computador/ Human Computer Interaction***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Maria Teresa Galvão Dias, TP-21 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***José Luís Cabral de Moura Borges, TP-21 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***No final desta unidade curricular, os estudantes deverão ser capazes de:***
***- compreender a importância da interação com o utilizador no desenvolvimento de sistemas interativos***

***- compreender os principais princípios de desenho de interação***
***- conceber um sistema interativo utilizando uma abordagem centrada no utilizador, baseada na recolha e definição de requisitos, e na construção e avaliação de proptótipos.***
***- fazer testes de usabilidade em sistemas já existentes, assim como propor e testar sugestões de melhoria.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***At the end of this course, students should be able to:***
***- understand the importance of interaction with the user in the development of interactive systems***
***- understand the main principles of interaction design***
***- design an interactive system using a user-centred approach based on gathering and defining requirements, and construction and evaluation of proptotypes.***
***- make usability tests on existing systems and propose and test suggestions for improvement.***

**3.3.5. Conteúdos programáticos:**
***1. Projeto de Concepção da Interacção.***
***2. Compreender e Conceitualizar a Interacção.***
***3. Compreender os Utilizadores.***
***4. Compreender como as interfaces afectam os utilizadores.***
***5. Interfaces e Interacção***
***6. O Processo de concepção da Interacção.***
***7. Identificar Necessidades de Definir Requisitos e construção de protótipos.***
***8. Introdução à Avaliação de Usabilidade.***

**3.3.5. Syllabus:**
***1. Interaction Design Project.***
***2. Understanding and conceptualizing interaction.***
***3.Understanding users.***
***4. Affective Aspects.***
***5. Interfaces and Interactions***
***6. The process of interaction design.***
***7. Identifying needs, establishing requirements and constructing prototypes.***
***8. Introduction to the evaluation of usability.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Na primeira parte da unidade curricular (pontos 1 a 5 dos conteúdos programáticos) são apresentados aos estudantes os conceitos básicos de interação, explicando a sua importância para o utilizador e mostrando a sua evolução ao longo do tempo. Na segunda parte da unidade curricular (pontos 6 e 7 dos conteúdos programáticos) é apresentada uma metodologia de desenho de sistemas interativos centrada no utilizador. São descritas diversas formas de recolha e definição dos requisitos do utilizador, processos de construção de protótipos e maquetes, assim como metodologias para a sua apresentação e avaliação envolvendo utilizadores. Na terceira parte da unidade curricular são apresentados vários métodos de avaliação de usabilidade: testes de usabilidade, avaliação analítica e estudos de campo.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***In the first part of the module (points 1-5 of the syllabus), the basic concepts of interaction are presented to the students, explaining their importance to the user and showing their evolution over time. In the second part (sections 6 and 7 of the syllabus), a user-centred methodology for designing interactive systems is presented. We describe various ways of gathering and defining user requirements, building prototypes and mock-ups, as well as methods for their evaluation with users.***
***In the third part of the module, several methods for usability evaluation are presented to students: usability testing, analytical evaluation and field studies.***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***A lecionação desta unidade curricular baseia-se nos seguintes componentes de ensino pelos docentes e de aprendizagem pelos estudantes:***
***A1. Aulas Teórico-Práticas***
***Durante as aulas teórico-práticas serão expostas e discutidas noções teóricas que deverão ser apreendidas pelos estudantes através de atividades de estudo e conceitualização. Estas aulas incluem também a exposição e discussão em grupo de casos de estudo e respetivas resoluções, os quais deverão ser objeto de estudo e experimentação com novos problemas por parte dos estudantes.***
***A2. PROJETO DE IPC***
***Projeto de análise de um sistema interativo e especificação da sua evolução***
***COMPONENTES DE AVALIAÇÃO***
***A avaliação baseia-se nos seguintes 2 componentes (P1 e P2):***

***P1. Projeto de grupo – avaliação distribuída (50%).***
***P1.1 – Proposta do projeto de grupo (10%)***
***P1.2 – Relatório do Projeto (30%).***
***P1.3 – Apresentação do Projeto (10%).***
***P2. Prova escrita individual – avaliação com exame final (50%).***

**3.3.7. Teaching methodologies (including assessment):**

***The teaching is based on the following teaching and learning elements:***
***A1. PRACTICAL CLASSES***
***In the practical classes, the theoretical ideas behind the module are presented, and students learn them by studying***
***and conceptualizing. These classes also include the presentation and disussion in groups of a number of case studies and their resolution, and the students are then presented with new problems to study and resolve.***
***A2. HUMAN-COMPUTER INTERACTION PROJECT***
***Students have to carry out a group project on the analysis of an interactive system and its evolution.***

***ASSESSMENT***
***Assessment will be based on the following components (P1 and P2):***
***P1. Group Project- continuous assessment (50%)***
***P1.1 – Group assignment proposal (10%)***
***P1.2- Project report (30%)***
***P1.3- Project presentation (10%)***
***P2. Individual written exam- 50%***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***Durante as aulas teórico-práticas são apresentados os conceitos básicos de interação, as metodologias de desenho e as principais técnicas de avaliação. Os estudantes são agrupados em equipas de 6 ou 7 elementos e são propostos dois casos práticos, trabalhados em momentos diferentes.***
***No primeiro caso é apresentada uma ideia de um sistema interativo, para o qual os estudantes terão de aplicar a metodologia de desenho centrada no utilizador. Deverão identificar necessidades e requisitos,***
***propor soluções baseadas em maquetes e protótipos. As propostas de cada grupo são apresentadas e discutidas na aula.***
***O segundo caso consiste na avaliação de um sistema interativo existente. Os estudantes deverão selecionar a técnica de avaliação mais apropriada e elaborar um relatório de avaliação no final da aula. A unidade curricular inclui ainda um projeto de grupo (4 ou 5 elementos), realizado fora das aulas, no qual os estudantes selecionam um sistema real, analisam-no do ponto de vista da interação, fazem testes de***
***usabilidade, propõem melhorias usando maquetes ou protótipo e testam as suas propostas. No final entregam um relatório e apresentam o trabalho em público.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***In the practical classes, the basics of interaction, the design methodologies and the main evaluation techniques are presented.***
***Students are grouped into teams of 6 or 7, and work on two case studies, at different times.***
***The first consists of an idea for an interactive system, for which students must apply a user-centred design. They should identify needs and requirements and propose solutions based on mock-ups and prototypes. The proposals from each group are presented and discussed in class.***
***The second case study consists of evaluating an existing interactive system. Students should select the most appropriate evaluation technique and prepare a report at the end of class.***
***The module also includes a group project (4 or 5 members) conducted outside the classroom, in which students select a real interactive system, analyse it from the point of view of interaction, do usability testing, using mock-ups or prototypes, propose improvements using mock-ups or a prototype and test their proposals. At the end, the students submit a report and present the work in public.***

**3.3.9. Bibliografia principal:**

***- Preece, J., Rogers, Y. & Sharp, H. (2007).Interaction Design – Beyond Human Computer Interaction (4rd Edition). NY. John Wiley & Sons.***
***- Budiu, R. & Nielsen, J. (2013). Mobile Usability. New Riders Press.***
***- Fonseca, M., Campos, D. & Gonçalves, D. (2012). Introdução ao Design de Interfaces. FCA.***
***- Krug, S. (2014). Don't Make Me Think Revisited: A Common Sense Approach to Web Usability. New Riders Press.***
***- Cooper, A., Reimann, R., Cronin, D. & Noessel (2014). About Face: The Essentials of Interaction Design (4rd Edition). NY. John Wiley & Sons.***
***- Shneiderman, B. (2002). Leonardo's Laptop: Human Needs and the New Computing Technologies. Cambridge MA. MIT Press.***
***- Norman, D. A. (2013). Design of Everyday Things: Revised and Expanded. London. MIT Press.***
***- Norman, D. (1999). The Invisible Computer. Cambridge MA. MIT Press.***

**Mapa IV - Laboratório de Engenharia de Software/Software Engineering Laboratory**

**3.3.1. Unidade curricular:**
***Laboratório de Engenharia de Software/Software Engineering Laboratory***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Nuno Honório Rodrigues Flores; TP-14 horas + PL-16 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Luís Filipe Pinto de Almeida Teixeira; PL-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***A - Aplicar um processo de Engenharia de Software ao desenvolvimento completo de um sistema de software real, ao longo do semestre, abrangendo a especificação de requisitos, arquitetura, implementação, integração, teste, documentação e demonstração.***
***B - Adquirir conhecimentos fundamentais e experiência sobre as práticas de desenvolvimento ágil de software: planeamento de iterações, testes unitários, refactoring, pattern-based design, autoria colectiva de código, programação em pares, integração contínua.***
***C - Utilizar ferramentas de desenvolvimento de software adequadas e que permitam o acompanhamento do desenvolvimento do produto durante todo o seu ciclo de vida.***
***D - Utilizar APIs e infraestruturas de software de grande escala, focando o desenvolvimento baseado em componentes e problemas de integração aplicacional.***
***E - Desenvolver trabalho colaborativo, integrando outros intervenientes do projeto em decisões de desenho e planeamento e delegar, negociar e rever estas decisões em grupo.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***A - To apply a Software Engineering process to the complete development of a real software system, throughout the module, covering the requirements specification, software architecture and design, coding, integration, testing, documentation and demonstration.***
***B - To acquire fundamental knowledge and experience about agile software development practices: iteration planning, unit tests, refactoring, pattern-based design, collective ownership, pair programming, continuous integration.***
***C - To use software development tools adapted to agile methods to enable the continuous monitoring and tracking of the project along its lifecycle.***
***D - To use large-scale APIs with class packages, to induct the component-based computing and related problems with application integration.***
***E - To undertake collaborative work, integrating other project participants in design decisions, planning, delegation, negotiation, and group review.***

**3.3.5. Conteúdos programáticos:**
***1 - Aplicação de processos de desenvolvimento de software, incluindo metodologias de planeamento de projeto.***
***2 - Identificação de requisitos, desenho, implementaçaão, integração, teste, documentação e demonstração de um sistema de software;***
***3 - Introdução às Metodologias Ágeis: valores, principios fundamentais, práticas ("Planning Game", "Small Releases", "Acceptance tests", "Unit-tests", "Test-first programming", "Simple Design", "Refactoring", "Design patterns") e exemplos de metodologias populares (Scrum e XP);***
***4 - Desenvolvimento de aplicações multi-camada baseadas em componentes de software;***
***5 - Abordagem a tecnologias nucleares nos domínios dos projetos em curso.***
***6 - Utilização de ambientes de desenvolvimento integrados (IDEs), ferramentas de modelação, teste, controlo de versões e configurações, documentação colaborativa de software e entrega de produto;***
***7 - Utilização de APIs de larga escala.***

**3.3.5. Syllabus:**
***1 - Application of software development processes, including project planning methodologies.***
***2 - Requirements capture and elicitation, design, implementation, testing, documentation and presentation of a software system.***
***3 - Introduction to Agile Methods: values, principles, practices (Planning, Small Releases, Acceptance tests, Unit-tests, Test-first programming, Simple Design, Refactoring, Design Patterns) and popular examples of agile methods (Scrum and XP).***
***4 - Design and development of multilayer applications based on software components and web services.***
***5 - Core technologies overview according to the developed projects domain areas.***

***6 - Use of tools and programming environments (IDEs), modelling tools, testing, source code control and configuration, collaborative documentation and deployment.***
***7 - Use of large-scale APIs.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***Os conteúdos programáticos (CPs) encontram-se alinhados com os objetivos de aprendizagem (OAs) pelo que abordam diretamente os conhecimentos e competências a serem adquiridos pelos estudantes. Este alinhamento pode-se verificar se se atentar nas seguintes afetações:***
***OA CP***
***_____***
***A 1,2***
***B 2,3***
***C 6***
***D 4,5,7***
***No caso do objetivo de aprendizagem E, todo o trabalho de projeto da unidade curricular será desenvolvido em ambiente de equipa, atribuíndo-se papeis de responsabilidade específicos aos seus intervenientes.***
***Em virtude da unidade curricular ser de índole laboratorial, a prática e aplicação dos conceitos abordados no programa, num projeto integrador e conciliador desses mesmos conceitos, permite a concretização e sedimentação das competências, num ambiente multi-disciplinar real, equiparado ao encontrado na Industria e nos contextos onde será exigido aos graduados a aplicação dessas mesmas competências.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The syllabus (SY) is aligned with the expected learning outcomes (LOs), focusing directly on the knowledge and skills required to achieve those outcomes. As such, the direct contributions from the syllabus towards a sucessful achivement of the learning outcomes can be stated as follows:***
***LO SY***
***----------***
***A 1,2***
***B 2,3***
***C 6***
***D 4,5,7***
***Learning outcome E derives from working as a multi-disciplinary team with specific roles and responsibilities throughout the whole project.***
***Through a team project-based learning environment, the students' required proficiency levels will be gradually acquired through practice, applying the designated methodologies, techniques and tools, over the course of the module.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***As aulas teórico-práticas consistem na exposição dos processos de desenvolvimento de software, ferramentas e tecnologias adoptadas na unidade curricular. As aulas de laboratório servirão para o acompanhamento e desenvolvimento do projeto, através do uso dos sistemas, linguagens e ferramentas escolhidos para a implementação da aplicação. O projeto está organizado em 5 iterações, com entregas de artefactos, e um peso respetivo na avaliação final.***
***-Iteração 0, 4 semanas, peso 20%, artefactos: visão do produto, user stories + interfaces utilizador, arquitetura, protótipo vertical funcional.***
***-Iterações 1-4, 8 semanas, peso 30%, artefactos: plano de desenvolvimento, user stories finais, estimativas, testes de aceitação, testes unitarios.***
***-Entrega final, peso 30%, artefactos: produto, código-fonte documentado, pitch+video+slides demo (5mins), documentação técnica final.***
***Os restantes 20% da nota consideram a plataforma de projeto montada e o desempenho individual dos estudantes.***

**3.3.7. Teaching methodologies (including assessment):**

***Practical classes are used for the presentation of software developing tools and of tools and techniques used in the mocule. Laboratory classes are used for the monitoring and development of the project, through the use of systems, languages and tools chosen for the implementation of the application. The development of the project will be divided into five iterations, with expected deliverables, contributing a percentage of the final grade:***
***-Iteration 0-20% 4 weeks, deliverables: vision, user stories + user interfaces, architecture, vertical functional prototype.***
***-Iterations 1-4 -30%,8 weeks,deliverables: development process plan, ready user stories, estimates, acceptance tests, unit tests.***
***-Final submission -30%,deliverables: product release (signup, download, etc.), documented source code repository, pitch +video+slides demo (5min), updated technical reports***
***The remaining 20% of the final grade is for an assessment of the project management platform and individual performance.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***Sendo esta unidade curricular de índole laboratorial, a componente teórico-prática das aulas será apenas de introdução de conceitos, conhecimentos, técnicas e ferramentas que os estudantes necessitarão de aprofundar durante as aulas laboratoriais. A metodologia de ensino com base num projeto permite que os estudantes, não só adquiram e pratiquem os conceitos abordados num ambiente real, mas igualmente desenvolvam o espirito de colaboração, cooperação e trabalho em equipa. O acompanhamento semanal da evolução dos projetos, quer presencial, quer através das ferramentas utilizadas para o efeito, permite uma aferição efetiva da evolução dos conhecimentos e competências adquiridas. A Engenharia de Software é, na sua aplicação mais corrente, o desenvolvimento de um projeto que começa na germinação de uma ideia ou conceito até ao culminar da concretização de um produto. Através da metodologia de ensino baseado em projetos, é possível não só adquirir os conceitos a aplicar, como aprofundá-los através da prática e integração dos mesmos, com um acompanhamento e monitorização regulares por parte dos docentes. Desta forma, adota-se uma filosofia pedagógica de "aprender, fazendo" que estimula a interiorização dos conhecimentos e expõe os estudantes aos problemas reais que surgem durante um projeto de engenharia de software. Através da demonstração pública dos resultados dos projetos, os estudantes são igualmente estimulados ao conceito de "produto" de software, apelando a uma componente inovadora e viabilizadora da possível atração de investidores interessados num produto real.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***Team project-based learning is the adopted methodology in this module. The theoretical and practical part are intended to introduce the concepts, techniques and tools to be used during the course of the project. The laboratory classes will enable the tutoring, supervision and monitoring of the project. Through a real-life software project scenario, the students will be able to acquire and develop all the intended practices and techniques that a software engineering project encompasses. From its inception to its final product delivery, the students will go through all the phases of development, working in teams and collaborating to achieve a common goal. This "learn by doing" pedagogical philosophy allows for a deeper understanding of the practices and exposes the students to the hurdles and difficulties a real software project might present. This methodology focuses on "product delivery", forcing the students to define a viable process to enable a fully functional, ready to use, final product release. On the way, quality standards need to be met, therefore best practices, methods and tools need to be used so that development enables the achievement of those standards. These quality metrics are set, monitored and enforced by the teachers. Innovation is endorsed through a product "pitch", evaluating if the final product will be prone to investment by potential stakeholders.***

**3.3.9. Bibliografia principal:**

***Sam Ruby, Dave Thomas, David Heinemeier Hansson,"Agile web development with Rails 4", The Pragmatic Programmers, 2013, ISBN: 0-9776166-3-0***
***Ken Schwaber, Mike Beedle, "Agile Software Development with Scrum", Pearson, 2001, ISBN-13: 978-0130676344***
***Kent Beck, Cynthia Andres, "Extreme Programming Explained: Embrace Change", Addison-Wesley, 2004, ISBN-13: 978-032127865***

**Mapa IV - Princípios e Paradigmas de Engenharia de Software/Software Engineering Fundamentals and Paradigms**

**3.3.1. Unidade curricular:**

***Princípios e Paradigmas de Engenharia de Software/Software Engineering Fundamentals and Paradigms***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**

***João Carlos Pascoal de Faria; TP-30 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**

***Nuno Honório Rodrigues Flores; TP-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**

***No final desta unidade curricular pretende-se que os estudantes:***
***1. Conheçam e saibam descrever criticamente os principais desafios, atividades e boas práticas para o desenvolvimento de software em larga escala;***
***2. Conheçam e saibam explorar os principais paradigmas e metodologias da engenharia de software, nomeadamente, métodos ágeis, métodos de desenvolvimento guiados por modelos e métodos formais;***
***3. Conheçam e saibam explorar as principais técnicas e ferramentas necessárias para executar e gerir as várias atividades do ciclo de vida do software.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**

***By the end of this module, it is intended that students:***
***1. Know and are able to critically describe the main challenges, activities and best practices for the development of large-scale software;***
***2. Know and are able to explore the main software engineering paradigms and methodologies, namely, agile methods, model-driven development methods, and formal methods;***
***3. Know and are able to explore the main techniques and tools needed to perform and manage the various activities in the software development lifecycle.***

**3.3.5. Conteúdos programáticos:**

***1. Problemática do desenvolvimento de software em larga escala;***
***2. Principais atividades e boas práticas do processo de desenvolvimento de software;***
***3. Métodos ágeis de desenvolvimento de software (incluindo Scrum e XP);***
***4. Métodos de desenvolvimento de software guiado por modelos (incluindo linguagens e ferramentas);***
***5. Métodos formais de desenvolvimento de software (incluindo linguagens e ferramentas);***
***6. Ferramentas integradas de gestão do ciclo de vida do software.***

**3.3.5. Syllabus:**

***1. Issues and challenges of large-scale software development;***
***2. Main activities and best practices in the software development process;***
***3. Agile software development methods (including Scrum and XP);***
***4. Model-driven software development methods (including languages and tools) ;***
***5. Formal methods of software development (including languages and tools) ;***
***6. Integrated software lifecycle management tools.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***Os 2 tópicos programáticos iniciais (problemática do desenvolvimento de software e atividades e boas práticas do processo de desenvolvimento de software) relacionam-se com o primeiro objetivo de aprendizagem, dotando assim os estudantes da capacidade de conhecer e saber descrever criticamente os principais desafios, atividades e boas práticas para o desenvolvimento de software em larga escala.***
***Os 3 tópicos programáticos seguintes, abordam diversos métodos de desenvolvimento de software, dotando assim os estudantes da capacidade de "conhecer e saber explorar os principais paradigmas e metodologias da engenharia de software, nomeadamente, métodos ágeis, métodos de desenvolvimento guiados por modelos e métodos formais" (2º objetivo de aprendizagem).***
***O último tópico programático, relacionado com ferramentas, permite dotar os estudantes da capacidade de conhecer e saber explorar as principais técnicas e ferramentas necessárias para executar e gerir as várias atividades do ciclo de vida do software (3º objetivo de aprendizagem).***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The two initial syllabus topics (software development issues and activities and best practices in the software development process) are related to the first learning objective, thus giving students the ability to know and critically describe the main challenges, activities and best practices in the development of large-scale software.***
***The following three syllabus topics address various software development methods, so providing students with the ability to know and be able to exploit the main paradigms and methodologies of software engineering, in particular agile methods, model-driven development methods and formal methods (2nd learning objective).***
***The last syllabus topics, related to tools, provides students with the ability to know and be able to exploit the main techniques and tools required to implement and manage the various activities of the software life cycle (3rd learning objective).***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***ENSINO***
***As aulas terão uma componente de exposição e discussão dos tópicos programáticos, e outra componente de realização de exercicíos práticos e pequenos projetos de exploração dos conceitos e técnicas e experimentação de ferramentas. Será reservada uma ou duas aulas para a apresentação de tópicos explorados pelos estudantes.***

***AVALIAÇÃO***
***1) Exame final individual (peso 40%);***
***2) Um trabalho em grupo de pesquisa e apresentação sobre um tema relacionado com o âmbito da unidade curricular (peso 15%);***
***3) Três pequenos trabalhos práticos em grupo, de exploração dos paradigmas estudados (peso 45%)***

**3.3.7. Teaching methodologies (including assessment):**

***TEACHING***
***The classes will comprise the presentation and discussion of topics and the development of practical exercises***

***and small projects by the students. Slots will be reserved for the presentation of special topics explored by the students.***

***ASSESSMENT***
***1) Final exam (individual; 40% of the final grade);***
***2) One group assignment researching and presenting a topic related to the module syllabus (15% of the final grade);***
***3) Three small group assignments exploring the studied paradigms (45% of the final grade).***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***A exposição e discussão dos tópicos programáticos nas aulas teóricas permitirá aos estudantes conhecer os desafios, atividades, boas práticas, paradigmas, metodologias, técnicas e ferramentas para o desenvolvimento de software em larga escala, atingindo assim parte dos objetivos de aprendizagem.***
***Adicionalmente, a realização de exercícios práticos e pequenos projetos dotará os estudantes da capacidade de explorar na prática as metodologias e ferramentas. O estudo a realizar sobre um tema na área, permitirá ao estudantes desenvolver o espírito crítico em relação a alguns dos tópicos abordados.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***The presentation and discussion of the topics in the lectures acquaints students with the challenges, activities, best practices, paradigms, methodologies, techniques and tools for the development of large-scale software, thus achieving part of the learning objectives.***
***The practical exercises and small projects also provide students with the ability to explore in practice the methodologies and tools. The study carried out on a topic in the area will enable students to develop critical thinking about some of the topics covered.***

**3.3.9. Bibliografia principal:**

***Sommerville, I. (2011). Software engineering (9th ed.). Addison-Wesley. ISBN: 9780137035151***
***Rubin, K. S. (2012). Essential Scrum: A Practical Guide to the Most Popular Agile Process. Addison-Wesley Professional. ISBN-13: 978-0137043293***
***Beck, K. (2004). Extreme Programming Explained: Embrace Change (2nd Ed.). Addison-Wesley Professional. ISBN: 978-0321278654***
***Brambilla, M., Cabot, J., Wimmer, M. (2012). Model-Driven Software Engineering in Practice. Morgan & Claypool Publishers. ISBN: 978-1608458820***
***Jackson, D. (2006). Software Abstractions. MIT Press. ISBN: 0-262-10114-9***
***Fitzgerald, J. (2005). Validated designs for object-oriented systems. Springer-Verlag. ISBN: 1-85233-881-4***

**Mapa IV - Segurança em Engenharia de Software/Security in Software Engineering**

**3.3.1. Unidade curricular:**

***Segurança em Engenharia de Software/Security in Software Engineering***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**

***Rui Filipe Lima Maranhão de Abreu; TP-21 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**

***Ana Cristina Ramada Paiva; TP-21 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**

***No final desta unidade curricular, pretende-se que os estudantes:***
***1. Reconheçam os problemas mais importantes de segurança em sistemas baseados em software, respetivas causas e consequências, e reconheçam a importância da utilização de boas práticas de engenharia de software para a prevenção, deteção e mitigação desses problemas;***
***2. Conheçam e sejam capazes de aplicar boas práticas genéricas e específicas para o desenvolvimento de sistemas de software seguros nas diversas fases do ciclo de vida do software - captura de requisitos, arquitetura e desenho, implementação, verificação e validação, distribuição, instalação e manutenção;***
***3. Conheçam e sejam capazes de usar processos e ferramentas de engenharia de software especialmente direcionados para questões de segurança;***
***4. Conheçam normas relevantes para o desenvolvimento de sistemas de software seguros.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**

***By the end of this module, it is intended that students:***

***1. Recognise the most important security problems of software-based systems, the respective causes and consequences, and recognise the importance of using good software engineering practices for the prevention, detection and mitigation of those problems;***
***2. Know and are able to apply generic and specific practices for developing secure software systems in the various phases of the software life cycle - requirements capture, architecture and design, implementation, verification and validation, deployment, installation and maintenance;***
***3. Know and are able to use software engineering processes and tools specially targeted at security issues;***
***4. Know relevant standards for the development of secure software systems.***

**3.3.5. Conteúdos programáticos:**
***1) Problemática do desenvolvimento de software seguro: problemas típicos de segurança no software (vulnerabilidades e ataques); impacto para os utilizadores e sociedade; importância do seguimento das boas práticas da engenharia de software para o desenvolvimento de software seguro; exemplos de casos reais;***
***2) Processos de engenharia de software com foco na segurança;***
***3) Engenharia de requisitos de segurança;***
***4) Desenho de sistemas de software seguros: princípios de desenho, estilos de arquitetura, padrões de desenho para segurança;***
***5) Programação segura: boas práticas e normas;***
***6) Verificação e validação de aspetos de segurança: ferramentas de análise estática; revisões e inspeções de segurança de software; testes de segurança; auditorias de segurança ao software; verificação formal;***
***7) Segurança na distribuição, instalação e manutenção de sistemas de software.***

**3.3.5. Syllabus:**
***1) Problems and challenges of secure software development: typical problems in software security (vulnerabilities and attacks); impact on users and society; importance of following software engineering best practices for secure software development; real world examples;***
***2) Software engineering processes with a security focus;***
***3) Security requirements engineering;***
***4) Design of secure software systems: design principles, architectural styles, secure design patterns;***
***5) Secure coding: best practices and standards;***
***6) Verification and validation of security aspects: static analysis tools; reviews and inspections of software security: security testing; software security audits; formal verification;***
***7) Security in the deployment and maintenance of software systems.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***Os conteúdos programáticos foram definidos em função dos objetivos e competências a serem adquiridos pelos estudantes nesta UC. Os conteúdos programáticos incluem as abordagens consideradas estado-da-arte em segurança em engenharia de software bem como técnicas específicas que conferem aos estudantes a capacidade de selecionar de forma crítica o método apropriado a utilizar para, na prática, resolver problemas de segurança na engenharia de aplicações informáticas. Ao mesmo tempo, os estudantes serão capazes de interpretar e analisar o comportamento dos algoritmos e os resultados obtidos pelos mesmos. Para atingir os objectivos de aprendizagem, a UC irá abordar não só processos e algoritmos, bem como ferramentas off-the-shelf.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***The syllabus was defined according to the objectives and competencies to be acquired by the students. The syllabus includes understanding state-of-the-art approaches to security in software engineering as well as specific techniques, which will make students capable of critically selecting the appropriate method to use to solve the engineering problem at hand. At the same time, students will be able to interpret and analyse the behaviour and performance of the algorithms and the results obtained by them. In order to achieve the proposal learning outcomes, the module will not only discuss processes and algorithms, but also off-the-shelf tools.***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***ENSINO***
***As aulas terão uma componente de exposição e discussão dos tópicos programáticos, e outra componente de realização de exercicíos práticos e pequenos projetos de aplicação dos conceitos e técnicas e experimentação de ferramentas. Será reservada uma ou duas aulas para a apresentação de tópicos explorados pelos estudantes.***

***AVALIAÇÃO:***
***A componente de avaliação contínua (50% da nota final) compreende:***
***- Participação nas aulas (5%);***
***- Trabalhos práticos (45%) (Trabalho de modelação e especificação de requisitos: 25%; Palestra sobre um tema de RE: 20%)***
***- Exame Final (50% da nota final)***

***Para aprovação é necessário um mínimo de 45% no exame final.***

**3.3.7. Teaching methodologies (including assessment):**

***TEACHING***
***The classes will comprise the presentation and discussion of topics and the development of practical exercises and small projects by the students. Slots will be reserved for the presentation of special topics explored by the students (having an article and presentation as output).***

***ASSESSMENT:***
***Continuous assessment (50% of the final grade) comprises:***
***- participation in the classes (5%);***
***- Practical work (45%): (Modelling and requirements specification: 25%; lecture on an RE topic: 20%);***
***- Final Exam (50% of the final grade).***
***For approval, a minimum grade of 45% in the final exam is required.***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***Esta unidade curricular inclui aulas de índole teórico-prática (3h/semana), que são divididas numa parte de caráter mais expositivo (1h) e noutra essencialmente de análise e implemtação prática dos conceitos teóricos estudados e projeto (2h).***
***A primeira parte contempla a explicação de conceitos, padrões ou métodos de abordagem, sempre com a apresentação de exemplos e demonstrações, seguida de uma parte de aplicação com a resolução de pequenos problemas no contexto da segurança de aplicações de software que exercitem e demonstrem alguns dos conceitos apresentados. Sempre que possível serão estudados casos e comprometimentos de segurança reais em aplicações reais.***
***A unidade curricular inclui também um trabalhos práticos que se extenderá durante o semestre e abordará os tópicos da segurança nas diferentes fases do desenvolvimento de software. Este trabalho é em geral a solução para um cenário empresarial significativo. Os trabalhos procuram sempre abranger as mais recentes tecnologias, preparando dessa forma os estudantes para o mercado de trabalho. Todas estas ações e métodos de aprendizagem, baseados sobretudo em fazer, mas sem esquecer a aprendizagem de conceitos e metodologias de aplicação geral, contribuem de forma decisiva para o atingir dos objetivos propostos e a aquisição das competências necessárias à exploração deste tipo de sistemas.***
***A avaliação dos estudantes servirá para a aferição da eficácia das metodologias de ensino desenvolvidas na observância dos objetivos da UC e, se necessário, no futuro poder-se-á realizar algumas correções nas metodologias de ensino.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***This module includes only practical classes (3h/week), split into one part that is more exposition oriented (1h) and another that is essentially the implementation of the concepts learned in practice as well as design (2h).***
***In the first part, the concepts and methodology in the syllabus are presented, followed by a second part of application, where some examples and demonstrations are offered and a number of small problems are solved, exercising and demonstrating the security in software engineering concepts advocated.***
***The module also includes one assignment of broader significance, requiring the design, implementation, deployment and demonstration of a complete and real system, paying special attention to security aspects. The assignment is frequently the solution for a significant real-life industrial scenario problem. The aim of the assignment is also to make the students use a state-of-the-art technological solution, thus preparing them for the real world. All these actions and learning methods, mainly hands-on oriented, but also focusing on learning the general concepts, patterns and methods in common use, contribute decisively to achieving the proposed objectives and the skills acquisition and practice needed for the exploitation of this kind of system.***

***The assessment process will help the efficacy of the teaching methodologies of this module to be understood and, if necessary, the necessary adjustments will be made in conformity with the feedback from the students.***

**3.3.9. Bibliografia principal:**

***- Seacord , R.C. (2013). Secure Coding in C and C++ (2nd Edition). Addison-Wesley Professional.***
***- Seacord , R.C. (2008). The CERT C Secure Coding Standard. Addison-Wesley Professional.***
***- McGraw, G. (2006). Software Security: Building Security In. Addison-Wesley Professional.***
***- Dowd, M., McDonald, J., & Schuh, J. (2006). The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities. Addison Wesley Professional.***
***- Davis, N. (2005). Secure Software Development Life Cycle Processes: A Technology Scouting Report. CMU/SEI-2005-TN-024, Software Engineering Institute.***
***- Mead, N.R., Hough, R., & Stehney II, T. (2005). Security Quality Requirements Engineering (SQUARE) Methodology. CMU/SEI-2005-TR-009, Software Engineering Institute.***
***- Seacord, R.C., Svoboda, D., & Togashi, K. (2009). Secure Design Patterns. Chad Dougherty, Kirk Sayre.***

***Papers publicados em conferências de referência na área de Segurança em Engenharia de Software.***

**Mapa IV - Seminários em Engenharia de Software /Software Engineering Seminars**

**3.3.1. Unidade curricular:**
***Seminários em Engenharia de Software /Software Engineering Seminars***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Ademar Manuel Teixeira de Aguiar (TP-30 horas)***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***Raul Fernando de Almeida Moreira Vidal (TP-12 horas)***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***O objetivo desta unidade curricular é apresentar sob a forma de seminários independentes um leque alargado de subtópicos de Engenharia de Software, que pela sua pertinência, em termos de investigação, inovação ou aplicação industrial, permitam aos estudantes preprarem-se melhor para a eventual realização de um trabalho de investigação cientifica ou aplicada numa área da Engenharia de Software.***
***No final do ciclo de estudos, os estudantes deverão ter uma visão alargada de tópicos e subtópicos relevantes na área de Engenharia de Software.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**
***The objective of this module is to use the independent seminar format to present a wide range of subtopics on Software Engineering which might be pertinent for research, innovation or industrial application purposes, and therefore may help students to become better prepared to do research work in the Software Engineering field.***
***By At the end of the module, students should have a wide view of topics and subtopics relevant in the Software Engineering field.***

**3.3.5. Conteúdos programáticos:**
***Tópicos de investigação e evoluções recentes do conhecimento em Engenharia de Software, nomeadamente:***
***- Engenharia de software baseada em modelos;***
***- Reutilização de software em larga escala (linhas de produtos de software, padrões e frameworks);***
***- Manutenção de software (compreensão, engenharia reversa);***
***- Processos de software e melhoria de processos (métodos ágeis, PSP/TSP/CMMI);***
***- Linguagens dinâmicas;***
***- Gestão de conhecimento em Projetos de Software;***

**3.3.5. Syllabus:**
***Research topics and recent development of knowledge in the field of Software Engineering, namely:***
***-Model-driven software engineering***
***- Software reuse on a large scale (software product lines, patterns, frameworks);***
***- Software maintenance (software understanding, reverse engineering);***
***- Software processes and process improvement (agile methods, PSP/TSP/CMMI);***
***- Dynamic languages;***
***- Software knowledge management.***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**
***A diversidade dos conteúdos abordados na UC permitirão alargar o conhecimento dos estudantes em tópicos diverisificados, especificos e pertinentes na área de Engenharia de Software, a selecionar em cada edição de acordo com o perfil dos estudantes inscritos e as suas preferências de investigação e desenvolvimento, preparando-os assim encontrarem e explorarem os seus tópicos de maior interesse, quer a nível pessoal, quer a nível organizacional.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**
***The diversity of the contents addressed in the module will enable students to have a wider knowledge of various specific topics that are pertinent to the Software Engineering field. The topics will be selected each year depending on the students' profiles and their research and development preferences, therefore preparing them to find and explore their own preferred topics, whether personal or organisational.***

**3.3.7. Metodologias de ensino (avaliação incluída):**
***As aulas serão dedicadas essencialmente à apresentação e discussão de tópicos seleccionados, sempre que***

***possível por convidados com experiência consolidada no tópico respectivo.***
***Cada estudante realizará um "survey" num tópico de investigação em engenharia e software, envolvendo análise bibliográfica, estudo de casos e experimentação. Cada estudante deve ainda analisar pelo menos um artigo científico indicado sobre cada tema abordado e produzir um pequeno texto de resumo e análise crítica do mesmo (1 a 2 páginas).***
***Tipo de Avaliação: Avaliação distribuída sem exame final.***
***Condições de Frequência: a nota mínima de 45% na componente de avaliação contínua.***
***Fórmula de avaliação:***
***-"survey" (60%)***
***-análise critica sobre cada tópico (40%)***

**3.3.7. Teaching methodologies (including assessment):**
***The classes will take the form of seminars on selected topics, presented by guest speakers with solid experience in the respective topic.***
***Each student will undertake a "survey" on one of the topics, including literature search, case studies and experiments. Each student will also write a scientific report about one topic and a 1 or 2 page summary on each topic presented.***
***Assessment: continuous assessment without final exam.***
***Attendance conditions: a minimum grade of 45% in the continuous assessment component.***
***Assessment weightinig:***
***-"survey" (60%)***
***-critical analysis of each topic (40%)***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**
***As aulas são coerentes com os objetivos da unidade curricular. Em particular, a transmissão de conhecimentos nas aulas, a realização de seminários sobre tópicos avançados e relevantes, discussão de casos de estudo nas aulas, e o aprofundar e apresentar um tema de Engenharia de Software.***
***Em particular, o trabalho de síntese e "survey" vai permitir aos estudantes ganharem um conhecimento mais aprofundado sobre o tópico respectivo, bem como adquirirem capacidade critica nesse tópico.***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**
***Classes are consistent with the objectives of the course. In particular the transmission of knowledge in the classroom, seminars on relevant advanced topics, discussion of case studies and the detailing and presentation of a topic on Software Engineering.***
***The survey and critical analysis assignments in particular will help students acquire a more solid knowledge about the respective topic, as well to gain critical analysis skills on that topic.***

**3.3.9. Bibliografia principal:**
***Bourque, P. & Fairley, R. E. (eds.) (2014). SWEBOK: Guide to the Software Engineering Body of Knowledge. Los Alamitos, CA: IEEE Computer Society Press. ISBN: 978-0-7695-5166-1***
***Sommerville, I. (2010). Software Engineering. Harlow, England: Addison-Wesley. ISBN: 978-0-13-703515-1***
***Serão ainda usados diversos artigos e documentos adequados aos temas concretos abordados nos seminários de cada ano letivo***

**Mapa IV - Teste, Verificação e Validação de Software/Software Testing Verification and Validation**

**3.3.1. Unidade curricular:**
***Teste, Verificação e Validação de Software/Software Testing Verification and Validation***

**3.3.2. Docente responsável (preencher o nome completo) e respectivas horas de contacto na unidade curricular:**
***Ana Cristina Ramada Paiva, TP-30 horas***

**3.3.3. Outros docentes e respectivas horas de contacto na unidade curricular:**
***João Carlos Pascoal de Faria, TP-12 horas***

**3.3.4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):**
***1) Familiarização com a terminologia usada em teste de software.***
***2) Conhecer e saber aplicar as técnicas principais de teste para controlo de qualidade de software.***
***3) Ser capaz de identificar e conhecer as questões relacionadas com a gestão de processos de teste de software.***

**3.3.4. Intended learning outcomes (knowledge, skills and competences to be developed by the students):**

***1) To be familiarised with the terminology used in software testing.***
***2) To acquire knowledge of the main techniques that may be applied in software quality control.***
***3) To be able to identify and understand issues related to the management of the software testing process.***

**3.3.5. Conteúdos programáticos:**

***1. Conceitos básicos de qualidade de software:***
***noção de qualidade; atributos e métricas de qualidade segundo a norma ISO 9126; fatores de qualidade; custos de qualidade; verificação e validação versus garantia de qualidade.***

***2. Teste de software: conceitos básicos e definições; técnicas de conceção de casos de teste (testes de caixa negra e de caixa branca); testes unitários e "test-driven development"; testes de integração; testes de sistema (teste através da interface para o utilizador, teste de desempenho, fiabilidade, disponibilidade, compatibilidade, etc.); teste baseado em modelos; testes de aceitação; automatização dos testes.***

***3. Inspeções, revisões e outras técnicas de análise estática de software.***

***4. Gestão do processo de testes: gestão de incidentes; reporte e seguimento de problemas; planeamento e documentação de testes; métricas relacionadas com testes***

**3.3.5. Syllabus:**

***1. Basic concepts of software quality:***
***notion of quality; attributes and quality metrics according to ISO 9126; quality factors; quality costs; verification and validation versus quality assurance.***

***2. Software testing: basic concepts and definitions; Test case design techniques (black box and white box testing); unit testing and "test-driven development"; integration testing; System tests (through the user interface, performance testing, reliability, availability, compatibility, etc.); Model-based testing; acceptance testing; Automation of testing.***

***3. Inspections, reviews and other static analysis software.***

***4. Management of the testing process: incident management; reporting and tracking problems; planning and documentation of tests; metrics related to testing***

**3.3.6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular:**

***O conteúdo programático está alinhado com os standards internacionais o que dá a familiarização adequada com a terminologia usada em testes de software no mundo.***
***Os módulos 2 e 3 dão uma visão alargada das principais técnicas de teste e controlo da qualidade para que os estudantes consigam escolher aquelas que se adaptam melhor em cada situação.***
***O modulo 4 dá o conhecimento necessário a quem quer optar por uma carreira ligada à gestão de projetos de teste e qualidade.***

**3.3.6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:**

***The syllabus is aligned with international standards, giving proper familiarity with the terminology used in software testing in the world.***
***Topics 2 and 3 give a broad overview of the main testing techniques and quality control so that students are able to choose the ones that best suit each situation.***
***Topic 4 provides the necessary knowledge for anyone choosing a career related to the management of testing and quality projects.***

**3.3.7. Metodologias de ensino (avaliação incluída):**

***As aulas serão divididas numa componente mais teórica, usada para exposição formal da matéria, acompanhada da apresentação de casos e sua discussão e uma parte mais prática onde os estudantes terão a oportunidade de realizar pequenos projetos e realizar um trabalho de exploração de ferramentas de teste. Este trabalho será dividido em duas fases: na primeira fase, os estudantes terão que preparar a apresentação de uma ferramenta de teste e elaborar um exercício prático de utilização; na segunda fase, os estudantes terão que produzir um relatório com um levantamento e comparação de ferramentas de teste de um determinado tipo.***
***Fórmula de cálculo da classificação final:***
***- (E) Exame final com peso de 50%. Nota mínima de 45% no exame.***
***Trabalhos práticos com peso de 50%:***
***- (A) aula prática: 25%;***
***- (P) trabalho de pesquisa: 25%.***
***Classificação Final = (E)*50% + (A)*25% + (P)*25%***

**3.3.7. Teaching methodologies (including assessment):**

***Classes will be divided in a more theoretical component, used for the formal explanation of the topic followed by the presentation and discussion of the cases, and a more practical component in which students will have the opportunity to conduct small projects and undertake conduct a test tool exploration assignment. This assignment will be divided into two phases: in the first, students will have to prepare the submission of a testing tool and work out a practical exercise for its use; in the second phase, students must produce a report with a survey and comparison of a given type of testing tool.***
***Assessment weighting:***
***- (E) Final exam with 50% weighting. Minimum score of 45% required.***
***Practical work with 50% weighting:***
***- (A) classroom practice: 25%;***
***- (P) research work: 25%.***
***Final Rating = (E) * 50% + (A) * 25% + (P) * 25%***

**3.3.8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:**

***A componente mais teórica permitirá introduzir os estudantes aos conceitos e terminologia usada em testes de software, nomeadamente, os conceitos básicos, inspeções, revisões e gestão do processo de teste (ponto 1 referido na secção 3.3.4). Haverá também uma componente de exercícios que permitirá praticar as diferentes técnicas de conceção de casos de teste (ponto 2 referido na secção 3.3.4). A componente mais prática permitirá que os estudantes pratiquem gestão de processos de teste de software e conheçam as ferramentas e abordagens de teste atuais e selecionar aquelas que são mais adequadas num determinado contexto (ponto 3 referido na secção 3.3.4). O conjunto de ferramentas a usar nas aulas cobrirá os vários níveis de teste: unitário, integração, sistema e aceitação (mencionados no ponto 2 referido na secção 3.3.5 deste documento).***

**3.3.8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes:**

***The more theoretical component will introduce students to the concepts and terminology used in software testing, in particular the basics concepts, inspection, review and management of the testing process (point 1 in section 3.3.4). There will be also exercises that will enable students to practise different test design techniques (point 2 in section 3.3.4). The more practical component will enable students to practise test process management and know the current testing tools and approaches, and select those that are most appropriate in a given context (point 3 in section 3.3.4). The set of tools to be used in class will cover the various test levels: unit, integration, system and acceptance (mentioned in paragraph 2 referred to in section 3.3.5 of this document).***

**3.3.9. Bibliografia principal:**

***Ammann, Paul; Offutt, Jeff; Introduction to software testing. 2008, ISBN: 0521880386***
***Mathur, Aditya P.; Foundations of software testing, 2nd Edition. 2013, ISBN: 9789332517660***
***Kent Beck; Test Driven Development: By Example. 2002, ISBN: 978-0321146533***
***Rick Mugridge, Ward Cunningham; Fit for Developing Software: Framework for Integrated Tests, 2005, Prentice Hall, 2005. ISBN: 978-0321269348***
***Utting, Mark; Legeard, Bruno; Practical Model-Based Testing: A Tools Approach. 2006, ISBN: 0123725011***
***Mitchell, Jamie L.; Black, Rex; Advanced Software Testing – Vol. 3, 2nd Edition, 2015, SBN: 9781937538644***

# 4. Descrição e fundamentação dos recursos docentes do ciclo de estudos

## 4.1 Descrição e fundamentação dos recursos docentes do ciclo de estudos

**4.1.1. Fichas curriculares**

**Mapa V - Ademar Manuel Teixeira de Aguiar**

**4.1.1.1. Nome do docente (preencher o nome completo):**

***Ademar Manuel Teixeira de Aguiar***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**

***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**

***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Ana Cristina Ramada Paiva**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Ana Cristina Ramada Paiva***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - António Miguel Pontes Pimenta Monteiro**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***António Miguel Pontes Pimenta Monteiro***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Carlos Manuel Milheiro de Oliveira Pinto Soares**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Carlos Manuel Milheiro de Oliveira Pinto Soares***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**

***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Associado ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Gil Manuel Magalhães de Andrade Gonçalves**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Gil Manuel Magalhães de Andrade Gonçalves***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Assistente convidado ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***34,4***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Hugo José Sereno Lopes Ferreira**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Hugo José Sereno Lopes Ferreira***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar convidado ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - João Carlos Pascoal de Faria**

**4.1.1.1. Nome do docente (preencher o nome completo):**

***João Carlos Pascoal de Faria***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - José Luís Cabral Moura Borges**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***José Luís Cabral Moura Borges***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Luís Filipe Pinto de Almeida Teixeira**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Luís Filipe Pinto de Almeida Teixeira***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Maria Teresa Galvão Dias**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Maria Teresa Galvão Dias***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Nuno Honório Rodrigues Flores**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Nuno Honório Rodrigues Flores***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Raul Fernando de Almeida Moreira Vidal**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Raul Fernando de Almeida Moreira Vidal***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Associado ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**Mapa V - Rui Filipe Lima Maranhão de Abreu**

**4.1.1.1. Nome do docente (preencher o nome completo):**
***Rui Filipe Lima Maranhão de Abreu***

**4.1.1.2. Instituição de ensino superior (preencher apenas quando diferente da Instituição proponente mencionada em A1):**
***<sem resposta>***

**4.1.1.3 Unidade Orgânica (preencher apenas quando diferente da unidade orgânica mencionada em A2):**
***<sem resposta>***

**4.1.1.4. Categoria:**
***Professor Auxiliar ou equivalente***

**4.1.1.5. Regime de tempo na Instituição que submete a proposta (%):**
***100***

**4.1.1.6. Ficha curricular do docente:**
**Mostrar dados da Ficha Curricular**

**4.1.2 Equipa docente do ciclo de estudos**

**4.1.2. Equipa docente do ciclo de estudos / Teaching staff of the study programme**

| Nome / Name | Grau / Degree | Área científica / Scientific Area | Regime de tempo / Employment link | Informação/ Information |
|---|---|---|---|---|
| Ademar Manuel Teixeira de Aguiar | Doutor | Ciência de Computadores / Engenharia de Software | 100 | Ficha submetida |
| Ana Cristina Ramada Paiva | Doutor | Engenharia de Software | 100 | Ficha submetida |
| António Miguel Pontes Pimenta Monteiro | Doutor | Engenharia Eletrotécnica e de Computadores | 100 | Ficha submetida |
| Carlos Manuel Milheiro de Oliveira Pinto Soares | Doutor | Ciência de Computadores | 100 | Ficha submetida |
| Gil Manuel Magalhães de Andrade Gonçalves | Mestre | Engenharia Electrotécnica e de Computadores – Sistemas | 34.4 | Ficha submetida |
| Hugo José Sereno Lopes Ferreira | Doutor | Engenharia de Software | 100 | Ficha submetida |
| João Carlos Pascoal de Faria | Doutor | Engenharia Informática / Engenharia de Software | 100 | Ficha submetida |
| José Luís Cabral Moura Borges | Doutor | Computer Science | 100 | Ficha submetida |
| Luís Filipe Pinto de Almeida Teixeira | Doutor | Engenharia Electrotécnica e de Computadores | 100 | Ficha submetida |
| Maria Teresa Galvão Dias | Doutor | Ciências da Engenharia | 100 | Ficha submetida |
| Nuno Honório Rodrigues Flores | Doutor | Engenharia Informática / Engenharia de Software | 100 | Ficha submetida |
| Raul Fernando de Almeida Moreira Vidal | Doutor | Engenharia Eletrotécnica | 100 | Ficha submetida |
| Rui Filipe Lima Maranhão de Abreu | Doutor | Ciência de Computadores / Engenharia de Software | 100 | Ficha submetida |
| **(13 Items)** | | | **1234.4** | |

***<sem resposta>***

## 4.2. Dados percentuais dos recursos docentes do ciclo de estudos

**4.2.1.Corpo docente próprio do ciclo de estudos**

#### 4.2.1. Corpo docente próprio do ciclo de estudos / Full time teaching staff

| Corpo docente próprio / Full time teaching staff | ETI / FTE | Percentagem* / Percentage* |
|---|---|---|
| Docentes do ciclo de estudos em tempo integral na instituição / Full time teachers: | 12 | 97.2 |

#### 4.2.2.Corpo docente do ciclo de estudos academicamente qualificado

#### 4.2.2. Corpo docente do ciclo de estudos academicamente qualificado / Academically qualified teaching staff

| Corpo docente academicamente qualificado / Academically qualified teaching staff | ETI / FTE | Percentagem* / Percentage* |
|---|---|---|
| Docentes do ciclo de estudos com o grau de doutor (ETI) / Teaching staff with a PhD (FTE): | 12 | 97.2 |

#### 4.2.3.Corpo docente do ciclo de estudos especializado

#### 4.2.3. Corpo docente do ciclo de estudos especializado / Specialized teaching staff

| Corpo docente especializado / Specialized teaching staff | ETI / FTE | Percentagem* / Percentage* |
|---|---|---|
| Docentes do ciclo de estudos com o grau de doutor especializados nas áreas fundamentais do ciclo de estudos (ETI) / Teaching staff with a PhD, specialized in the main areas of the study programme (FTE): | 12 | 97.2 |
| Especialistas, não doutorados, de reconhecida experiência e competência profissional nas áreas fundamentais do ciclo de estudos (ETI) / Specialists, without a PhD, of recognized professional experience and competence, in the main areas of the study programme (FTE): | 0.3 | 2.4 |

#### 4.2.4.Estabilidade do corpo docente e dinâmica de formação

#### 4.2.4. Estabilidade do corpo docente e dinâmica de formação / Teaching staff stability and tranning dynamics

| Estabilidade e dinâmica de formação / Stability and tranning dynamics | ETI / FTE | Percentagem* / Percentage* |
|---|---|---|
| Docentes do ciclo de estudos em tempo integral com uma ligação à instituição por um período superior a três anos / Full time teaching staff with a link to the institution for a period over three years: | 12 | 97.2 |
| Docentes do ciclo de estudos inscritos em programas de doutoramento há mais de um ano (ETI) / Teaching staff registered in a doctoral programme for more than one year (FTE): | 0.3 | 2.4 |

## 4.3. Procedimento de avaliação do desempenho

**4.3. Procedimento de avaliação do desempenho do pessoal docente e medidas para a sua permanente actualização:**

***O artigo 74.º-A do Estatuto da Carreira Docente Universitária, Decreto-Lei n.º 205/2009, de 31 de Agosto, determina que os docentes estão sujeitos a um regime de avaliação do desempenho constante de regulamento a aprovar por cada instituição de ensino superior.***

***Na Universidade do Porto, o regulamento para a avaliação do desempenho dos docentes encontra-se publicado em Diário da República, 2ª série, n.º 154, de 10 de Agosto de 2010, despacho n.º 12912/2010. No regulamento de avaliação da FEUP (despacho n.º 5096/2012) as vertentes investigação, ensino, de transferência de conhecimento e de gestão universitária são explicitamente valoradas e avaliadas. Complementarmente estabeleceram-se procedimentos para avaliar o processo de ensino-aprendizagem, através da realização de inquéritos pedagógicos que se realizam no final de cada semestre letivo. Os resultados dos inquéritos são incorporados no relatório de unidade curricular, que funciona como um instrumento de controlo e avaliação da qualidade do serviço docente, reunindo toda a informação relevante sobre o funcionamento de uma unidade curricular. Os relatórios são analisados pelo diretor do respetivo ciclo de estudos e, nos casos mais relevantes, pelo Conselho Pedagógico da FEUP. São ainda usados para a atribuição de prémios de incentivo pedagógico aos docentes.***

***A FEUP em parceria com a Faculdade de Psicologia e Ciências da Educação (FPCEUP) criou em 2008 o Laboratório de Ensino e Aprendizagem (LEA), cujo objetivo é melhorar a qualidade do ensino e aprendizagem, através de projetos e de iniciativas de formação que melhorem o desempenho pedagógico e promovam o sucesso escolar. Os projetos presentemente em curso são os seguintes:***

***• "De par em par" é uma ação de formação multidisciplinar, voluntária e de confidencialidade garantida que consiste na observação de aulas em parceria. A observação de aulas baseada no conceito de amigo crítico (observação de pares) recorre à confiança do docente observado perante os seus pares para obter uma***

***observação da sua prática pedagógica e aumentar a sua sensibilidade pedagógica, tanto na posição de observado como na de observador.***
***• "Assessorias Pedagógicas" tem com objetivo melhorar a qualidade do ensino e aprendizagem na FEUP, analisando os resultados dos inquéritos pedagógicos e o historial de sucesso escolar. São efetuados estudos caso a caso que procuram identificar as razões que os explicam e propor medidas que melhorem globalmente os índices de desempenho.***
***Anualmente é feito um levantamento das necessidades de formação dos recursos humanos da Universidade do Porto, sendo disponibilizadas ações de formação para o pessoal docente, entre os quais se destacam as seguintes áreas de formação: Formação de Professores / Formadores e Ciências da Educação; Biblioteconomia; Ciências Informáticas.***

**4.3. Teaching staff performance evaluation procedures and measures for its permanent updating:**
***Article 74A of the Statute of the University Teaching Profession, Decree-Law No. 205/2009 of 31 August, provides that teachers are subject to a performance assessment system enshrined in regulations to be approved by each institution of higher education..***
***At the University of Porto, the regulation for teacher's performance assessment is published in the Official Gazette, 2nd Series, No. 154, of 10 August 2010, Order No. 12912/2010.***
***FEUP's assessment regulation (Order No. 5096/2012) explicitly values and evaluates four components: research, teaching, knowledge transfer and university management. It also establishes procedures to assess the process of teaching and learning by conducting educational surveys that take place at the end of each semester. Survey results are incorporated into the course report, as an instrument of control and evaluation of the quality of the teaching service, bringing together all relevant information on the operation of a course. The reports are reviewed by the course director and, in the most relevant cases, FEUP's Academic Council. These reports are also used for attributing educational incentive awards to teachers.***

***Through a partnership with the Faculty of Psychology and Educational Sciences (FPCEUP), FEUP founded, in 2008, the Teaching and Learning Laboratory (LEA), which aims at improving the quality of teaching and learning through projects and training initiatives. It promotes increasing educational performance and academic success. The following projects are currently underway:***
***•"De par em par" (roughly meaning "in pairs") is a multidisciplinary, voluntary training action, whose participant's confidentiality is guaranteed, and which consists of observing lessons together with a colleague. Class observation based on the concept of a critical friend (peer observation) relies on the trust the observed teachers have in their peers to provide a sound opinion on their pedagogical practice, and so enhance their pedagogical sensitivity, as both observer and observed.***
***•"Pedagogical consultants" has the goal of improving the quality of teaching and learning in FEUP, analysing the results of educational surveys and cases of success in the school's History. Per-case studies are made, trying to identify the reasons that explain those results and proposing measures aimed at improving overall performance indices.***

***At the University of Porto, human resources training needs are surveyed every year. Consequently, training activities are provided to the teaching staff, among which we highlight the following areas of training: Teacher Training/Trainers and Educational Sciences; Library Economics; Computer Sciences.***

## 5. Descrição e fundamentação de outros recursos humanos e materiais

**5.1. Pessoal não docente afecto ao ciclo de estudos:**
***Entre o pessoal não docente afeto à lecionação do ciclo de estudos, uma parte encontra-se ligado ao Departamento de Engenharia Informática (DEI), outros ao Centro de Informática Prof. Correia de Araújo (CICA) e outra parte está ligado aos Serviços Académicos (SERAC).***
***Do pessoal não docente do DEI, dois elementos (dos quatro no total) estarão parcialmente afetos ao MESW, garantindo as atividades de administração, gestão e secretariado. O pessoal do CICA, 65 no total, dará apoio de laboratório, a tarefas de ensino, investigação e serviços, no âmbito das atividades a desenvolver no MESW.***
***Os SERAC garantem as atividades no âmbito da administração e apoio na gestão de ciclo de estudos e cursos; a área do acesso e certificação; a área de gestão de estudante de acordo com as instruções tutelares e as diretivas dos Órgãos de Gestão. Os SERAC contam com 16 pessoas a tempo inteiro, que dão apoio transversal a todos os ciclos de estudos da Faculdade de Engenharia.***

**5.1. Non teaching staff allocated to the study programme:**
***The non-teaching staff allocated to MESW come from three areas: the Department of Information Engineering (DEI), Correia Araújo Informatics Centre (CICA) and Academic Services (SA). From the non-teaching staff at DEI, two (out of four) will be partially allocated to the MESW, dealing with the administrative, management and secretarial tasks. The CICA staff, a total of 65, will provide support to laboratories, educational tasks, research and services, within the activities to be undertaken at MESW.***
***The SA guarantee: administration activities and study course management support; service access and***

***certification; student management under the guardianship of instructions and directives fthe Governing Bodies. The SA have 20 full-time staff, who give cross support to all courses at the Faculty of Engineering.***

**5.2. Instalações físicas afectas e/ou utilizadas pelo ciclo de estudos (espaços lectivos, bibliotecas, laboratórios, salas de computadores, etc.):**

***As instalações físicas afetas ao Mestrado de Engenharia de Software incluem:***
***Biblioteca - Gabinetes de Estudo Individual (1 lugar, 16 salas); Sala de leitura (11 salas).***
***Sala de Aula - Anfiteatro (53 lugares, 10 salas)***
***Sala de Aula - Computadores (16-20 lugares, 3 salas)***
***Sala de Aula - Computadores (24-27 lugares, 14 salas)***
***Sala de Aula - Teórico Práticas (19-28 lugares, 35 salas)***
***Laboratório de Projeto - Acesso Livre (20 lugares, 1 sala)***
***Salas de Exame (44-89 lugares, 12 salas)***
***Laboratório de Investigação - Engenharia de Software (24 lugares, 1 sala)***

**5.2. Facilities allocated to and/or used by the study programme (teaching spaces, libraries, laboratories, computer rooms, etc.):**

***The physical facilities allocated to the Software Engineering Master include:***
***Library - Individual Study Offices (16 rooms, 1 seat per room)***
***Library - Reading room (11 rooms)***
***Classroom - Amphitheater (10 rooms, 53 seats per room)***
***Classroom - with Computers (3 rooms, 16-20 seats per room)***
***Classroom - with Computers (14 rooms, 24-27 seats per room)***
***Classroom - for both lectures and practical classes (35 rooms, 24-27 seats per room)***
***Project Lab - Open Access (1 room, 20 seats)***
***Examination rooms (12 rooms, 44-89 seats per room)***
***Research Lab - Software Engineering (24 seats)***

**5.3. Indicação dos principais equipamentos e materiais afectos e/ou utilizados pelo ciclo de estudos (equipamentos didácticos e científicos, materiais e TICs):**

***Os principais equipamentos e materiais afetos ao ciclo de estudos incluem:***
***Salas de Aula - Computadores Pessoais, Projetores de Vídeo, Equipamentos de Rede.***
***Laboratório de Projeto, Acesso Livre – Computadores.***
***Biblioteca - Editoras de Revistas Científicas; Bases de Dados Bibliográficas; Enciclopédias e Dicionários; Editoras de e-Books; Bibliotecas Repositórios Digitais; Livros de Texto; Revistas em papel; Títulos com acesso on-line.***
***Centro de Informática - Pontos de Acesso à Rede (cabo); Pontos de Acesso à Rede (wireless); Impressoras em rede; Servidores; Routers; Switches.***
***Cluster com interligação em GRID (64 processadores)***
***Labs. de Investigação Informática – Computadores.***

**5.3. Indication of the main equipment and materials allocated to and/or used by the study programme (didactic and scientific equipments, materials and ICTs):**

***The main equipment and materials allocated to the course include:***
***Classrooms:***
***•Personal Computers***
***•Video Projectors***
***•Network Equipment.***
***Project Lab:***
***•Computers***
***Library:***
***•Access to Scientific Journals***
***•Bibliographic Databases***
***•Encyclopedias and Dictionaries***
***•e-Books,***
***•Digital Repositories and Libraries***
***•Text Books***
***•Hardcopy Magazines***
***•Online articles.***
***Informatics Centre:***
***•Cable and wireless network access points***
***•Network printers***
***•Servers, routers***
***•Switches***
***•Grid computing (64 processors)***
***•Computers in research labs.***

# 6. Actividades de formação e investigação

## Mapa VI - 6.1. Centro(s) de investigação, na área do ciclo de estudos, em que os docentes desenvolvem a sua actividade científica

**6.1. Mapa VI Centro(s) de investigação, na área do ciclo de estudos, em que os docentes desenvolvem a sua actividade científica / Research Centre(s) in the area of the study programme, where the teachers develop their scientific activities**

| Centro de Investigação / Research Centre | Classificação (FCT) / Mark (FCT) | IES / Institution | Observações / Observations |
|---|---|---|---|
| INESC TEC - INESC Technology and Science | Excelente/Excellent | Instituto de Engenharia de Sistemas e Computadores do Porto (INESC Porto/FE/UP) | Resultados da última avaliação das unidades de I&D em 2013 |
| Centro de Investigação em Sistemas e Tecnologias (Systec - Instituto de Sistemas e Robótica) | Muito Bom/Very Good | Faculdade de Engenharia da Universidade do Porto (FE/UP) | Resultados da última avaliação das unidades de I&D em 2013 |

## Perguntas 6.2 e 6.3

**6.2. Mapa resumo de publicações científicas do corpo docente do ciclo de estudos, na área predominante do ciclo de estudos, em revistas internacionais com revisão por pares, nos últimos cinco anos (referenciação em formato APA):**

**http://www.a3es.pt/si/iportal.php/cv/scientific-publication/formId/9a5fda97-f056-f1de-d5f1-55fbf7c5af08**

**6.3. Lista dos principais projetos e/ou parcerias nacionais e internacionais em que se integram as actividades científicas, tecnológicas, culturais e artísticas desenvolvidas na área do ciclo de estudos:**

***O corpo docente do MESW participa e/ou lidera vários projetos de investigação e desenvolvimento e tem diversas relações internacionais.***

***Alguns projetos:***

***• AutoSeer - Automatic generation of Test Oracles for Software Error Detection***
***• PBGT - Pattern-based GUI testing***
***• AIMS - Accelerated Improvement Method Support***
***• TOCE - Testing Online Collaborative Editing***
***• GZoltar - A graphical, semi-automated debugger***
***• ReBORN - Innovative Reuse of modular knowledge Based devices and technologies for Old, Renewed and New factories, (FP7)***
***• SELSUS - Innovative strategies for Renovation and Repair in Manufacturing systems, (FP7)***

***Algumas parcerias:***

***• Universidades de Lisboa, Coimbra e Minho, São Paulo, Federal de Pernambuco, Carnegie Mellon, Graz, Sheffield, Luxemburg, Delft, Texas at Dallas, South Australia, Naples Federico II, Milano***
***• Palo Alto Research Center***
***• Microsoft Research***
***• Institute of Science and Technology (ESEO)***
***• Google***
***• Facebook.***

**6.3. List of the main projects and/or national and international partnerships, integrating the scientific, technological, cultural and artistic activities developed in the area of the study programme:**

***MESW faculty members are participants and/or leaders in several research and development projects and have several international relationships with other institutions.***

***Some projects:***

***•AutoSeer - Automatic generation of Test Oracles for Software Error Detection***
***•PBGT - Pattern-based GUI testing***
***•AIMS - Accelerated Improvement Method Support***
***•ESWT - Testing Online Collaborative Editing***
***•GZoltar - A graphical, semi-automated debugger***
***•Reborn - Innovative Reuse of modular knowledge based devices and technologies for Old, New and Renewed factories (FP7)***
***•SELSUS - Innovative strategies for Renovation and Repair in Manufacturing systems (FP7)***

***Some partnerships:***

***•Universities of Lisbon, Coimbra and Minho, São Paulo, Federal de Pernambuco, Carnegie Mellon University, Graz, Sheffield, Luxemburg, Delft, Texas at Dallas, South Australia, Naples Federico II, Milano***
***•Palo Alto Research Center***
***•Microsoft Research***
***•Institute of Science and Technology (ESEO)***
***•Google***

***•Facebook***

## 7. Actividades de desenvolvimento tecnológico e artísticas, prestação de serviços à comunidade e formação avançada

**7.1. Descreva estas actividades e se a sua oferta corresponde às necessidades do mercado, à missão e aos objetivos da instituição:**

***No contexto dos grupos de investigação da FEUP e das unidades de I&D associadas, existe trabalho de desenvolvimento tecnológico e de prestação de serviços para entidades externas do setor público e privado relacionada com o MESW.***
***Na formação avançada, a FEUP cobre áreas do MESW com o programa de doutoramento ProDEI, com a participação no MAPi (Doutoramento em Informática), e com cursos curtos de formação, por exemplo, "Testes de software" e "Hands-On Design Patterns".***
***As unidades curriculares (UCs) do MESW poderão ser oferecidas a pessoas externas à UP, como "Unidades Curriculares Singulares", e um sub-conjunto de Ucs poderá ser fornecido a profissionais proporcionando um aprofundamento e atualização de conhecimentos nas áreas do MESW.***
***A FEUP tem muitas iniciativas com impacto na comunidade nomeadamente a JuniFEUP, G.A.S.Porto, e a EpDAH, com atividades nacionais e internacionais de cooperação nas quais se espera que os estudantes no MESW possam vir a colaborar.***

**7.1. Describe these activities and if they correspond to the market needs and to the mission and objectives of the institution:**

***MESW related work on technological development and services provided to external entities from both public and private sectors is also undertaken within the context of FEUP research groups and associated R&D units.***

***In advanced training, FEUP covers MESW domain areas through the ProDEI doctoral programme, participation in the MAPi doctoral programme, and through short training courses, e.g. "Software testing" and "Hands-On Design Patterns".***
***MESW modules may also be offered to people outside the UP, as "Stand alone modules" and a sub-set of modules may be provided to professionals, allowing a deepening and updating of knowledge in the MESW areas.***

***FEUP has many initiatives that have an impact on the community, including JuniFEUP, GASPorto, and EpDAH, with national and international cooperation activities in which it is expected that students of MESW are likely to collaborate.***

## 8. Enquadramento na rede de formação nacional da área (ensino superior público)

**8.1. Avaliação da empregabilidade dos graduados por ciclos de estudos similares com base nos dados do Ministério da Economia:**

***Segundo dados da DGEEC (Direção-Geral das Estatísticas da Educação e Ciência) para o Ensino Superior, no agregado de todos os mestrados (13) na área da Engenharia Informática, dos 1357 graduados de 2011 a 2013, estiveram inscritos nos centros de emprego apenas 40 (3%), dos quais 23 em busca de 1º emprego e 17 em busca de novo emprego. Destes 40, 26 obtiveram emprego em menos de 6 meses, e 4 em menos de um ano. Assumindo que os restantes 10 ainda se encontram desempregados, tal equivale a uma taxa de desemprego na ordem dos 0,7%. Segundo um relatório (Fev.2014) da e-skills in Europe (iniciativa da Comissão Europeia), estavam em atividade em 2012, na Europa, cerca de 7.400.000 profissionais na área das TIC, abrangendo a quase totalidade dos setores da Economia, concluindo-se existir um estado de "total empregabilidade". Quanto à oferta de emprego em Portugal, um estudo pela mesma fonte (Jan.2014), prevê que, em 2020, a procura por graduados nas áreas das TIC seja na ordem dos 15000.***

**8.1. Evaluation of the graduates' employability based on Ministry of Economy data:**

***According to data collected by the DGEEC (Direção-Geral das Estatísticas da Educação e Ciência) regarding employability, from a total of 1357 graduates (from 2011 to 2013) in all the masters degrees (13) in the Informatics Engineering domain, only 40 registered at job centres (23 applying for a first job and 17 for a new job). 26 got a job within 6 months, and 4 were employed within a year. Assuming the remaining 10 are still unemployed, it gives an unemployment rate of 0.7%. A February 2014 report from the e-skills in Europe initiative (European Commission), stated that the ICT workforce in Europe in 2012 comprised 7.4 million workers, spread throughout almost every industry in the economy and not just the ICT industry sector. It seems reasonable to assume that almost full***

***employment of this occupational group exists in Europe. Another report (Jan. 2014) from the same source predicts that, by 2020, in Portugal alone, the excess demand for ICT practitioner jobs will top 15,000.***

**8.2. Avaliação da capacidade de atrair estudantes baseada nos dados de acesso (DGES):**

***O MESW visa atrair estudantes nacionais e estrangeiros com as Unidades Curriculares oferecidas em Inglês e com o estabelecimento de parcerias internacionais para fomentar a mobilidade de estudantes, docentes e investigadores, como forma de melhorar continuamente o curriculum, alinhado com as melhores práticas e conhecimento atual.***

***Pretende-se efetuar uma candidatura ao programa Erasmus Mundus Joint Master Degrees. Neste contexto, já está a ser negociada uma parceria com a Universidade de São Paulo e outras em análise (e.g., ESEO - Institute of Science and Technology, França).Outras parcerias serão analisadas.***

***Os dados da DGES indicam que a procura de cursos de mestrado integrado (5 anos) na área de Informática (âmbito mais largo que o MESW,) é superior no norte de Portugal:o MIEIC (FEUP-Porto) teve mais de 300 candidatos em primeira opção, na 1ª fase, nos dois últimos anos enquanto que os Mestrados similares de Coimbra (MEI) e Lisboa (MIEI) tiveram 100 e 200 candidatos respetivamente.***

**8.2. Evaluation of the capability to attract students based on access data (DGES):**

***MESW targets Portuguese and foreign students as potential applicants, and its modules are offered in English. Moreover, through the establishment of international partnerships, it fosters mobility for students, researchers and faculty, as a way of continuously improving the course's curriculum, aligned with the best practices and state of the art.***

***As part of a future application to the Erasmus Mundus Joint Master Degrees programme, a partnership is being negotiated with the Universidade de São Paulo, while further partners are being analysed (e.g. ESEO - Institute of Science and Technology, France). Other partnerships will follow.***

***According to the DGES, demand for MSc courses (5-year degrees) in the Informatics area (a wider scope than MESW) is higher in Northern Portugal: During the last two years, MIEIC (FEUP - Porto) had over 300 first choice candidates in the first stage of enrolment, whereas similar courses at Coimbra and Lisbon had 100 and 200 candidates, respectively.***

**8.3. Lista de eventuais parcerias com outras instituições da região que lecionam ciclos de estudos similares:**

***O MESW pretende estabelecer diversas parcerias e acordos bilaterais ERASMUS para mobilidade de estudantes com instituições Europeias. Pretende-se estabelecer também acordos similares (aos estabelecidos com a Europa) com o Brasil e USA.***

***Neste contexto, já está a ser negociada uma parceria com a Universidade de São Paulo (USP) no Brasil e outras em análise (por exemplo, ESEO - Institute of Science and Technology, França). Outras parcerias serão analisadas com vista à internacionalização. Neste contexto, ir-se-á começar pelas parcerias já existentes no seio do corpo docente do ciclo proposto das quais se listam algumas na secção 6.3 deste documento.***

***Com vista à internacionalização, está prevista uma candidatura ao programa Erasmus+, nomeadamente, ao Erasmus Mundus Joint Master Degree.***

***Para além destas parcerias poderá haver colaboração docente com outras faculdades da UP, nomeadamente, Faculdade de Ciências.***

**8.3. List of eventual partnerships with other institutions in the region teaching similar study programmes:**

***The MESW intends to establish a number of partnerships and bilateral agreements for ERASMUS student mobility with European institutions. It is also intended to establish similar agreements (to those established with Europe) with Brazil and the USA.***

***In this context, a partnership is being negotiated with the University of São Paulo (USP) in Brazil and others are being analysed (e.g. ESEO - Institute of Science and Technology, France). Other partnerships will be analysed with a view to internationalisation. The course starts, therefore, with the existing partnerships within the faculty of the proposed course, some of which are listed in Section 6.3 of this document.***

***With a view to internationalisation, an application is expected to be made to the Erasmus+ programme, in particular the Erasmus Mundus Joint Master Degree.***

***In addition to these partnerships, there may be teaching collaboration with other faculties of UP, particularly the Science Faculty.***

## 9. Fundamentação do número de créditos ECTS do ciclo de estudos

**9.1. Fundamentação do número total de créditos ECTS e da duração do ciclo de estudos, com base no determinado nos artigos 8.º ou 9.º (1.º ciclo), 18.º (2.º ciclo), 19.º (mestrado integrado) e 31.º (3.º ciclo) do Decreto-Lei n.º 74/2006, de 24 de Março:**

***A estrutura curricular do MESW cumpre os requisitos definidos no Decreto-Lei nº 74/2006 de 24 de março, na redação dada pelo Decreto-Lei nº 115/2013, de 7 de agosto para adequação dos ciclos de estudos aos princípios do***

***Processo de Bolonha, nomeadamente os requisitos definidos no artigo 18.º para ciclos de estudos conducentes ao grau de mestre.***
***De facto, o MESW está configurado como um mestrado com a duração de 4 semestres correspondentes a um total de 120 créditos ECTS, cumprindo assim o estipulado no número 1 do referido artigo. O ciclo de estudos integra: a) uma parte curricular, constituída por um conjunto organizado de unidades curriculares, a que correspondem 72 créditos ECTS do ciclo de estudos; b) uma dissertação de natureza científica sobre um tema da área de conhecimento do ciclo de estudos, realizada em ambiente académico ou misto (empresarial e académico), original e especialmente realizada para este fim, a que correspondem 48 créditos ECTS do ciclo de estudos.***

**9.1. Justification of the total number of ECTS credits and of the duration of the study programme, based on articles no.8 or 9 (1st cycle), 18 (2nd cycle), 19 (integrated master) and 31 (3rd cycle) of Decreto-Lei no. 74/2006, March 24th:**
***The curriculum structure of MESW meets the requirements set out in Decree-Law 74/2006 to adapt the courses of study to the principles of the Bologna Process, including the requirements set out in Article 19 for courses leading to the Master's Degree.***
***In fact, the MESW is configured as a Master's Degree with a duration of 4 semesters corresponding to a total of 120 ECTS credits, thus fulfilling the provisions of paragraph 1 of that article. The course includes: a) a curricular component, consisting of an organised set of modules corresponding to 72 ECTS course credits; b) A scientific dissertation on a topic of knowledge in the area of the course, conducted in an academic or mixed environment (business and academic), which is unique and specially undertaken for this purpose, which corresponds to 48 ECTS course credits.***

**9.2. Metodologia utilizada no cálculo dos créditos ECTS das unidades curriculares:**
***O ciclo de estudos foi estruturado de forma a que o tempo expectável de trabalho exigido aos estudantes fosse igual em cada semestre do ciclo de estudos (810 horas por semestre–30 ECTS). No 1º ano cada semestre compreende 5 unidades curriculares, tendo sido atribuídos 6 ECTS a cada uma delas, dado ser expectável um tempo de trabalho de 162 horas.***
***O 2º ano inclui 2 unidades curriculares com um tempo expectável de trabalho de 162 horas para as duas unidades curriculares de 6 ECTS e 1296 para a dissertação com 48 ECTS.***
***O cálculo de créditos ECTS foi efetuado de acordo com o regulamento de Aplicação do Sistema de Créditos Curriculares aos Cursos Conferentes de Grau da UP que define que um crédito corresponde a 27 horas de trabalho do estudante.***

**9.2. Methodology used for the calculation of the ECTS credits of the curricular units:**
***The course has been structured so that the expected working time required of students is equal in each half of the course (810 hours per semester, 30 ECTS). In the 1st year, each semester comprises five modules, awarded 6 ECTS each, and involving 162 hours of working time.***
***The 2nd year includes 2 modules with an expected 162 hours of working time for the two modules, and awarded 6 ECTS each, and 1296 hours for the dissertation, awarded 48 ECTS (which will continue throughout the second semester).***
***The ECTS credits calculation was made according to the Curriculum Credits Application System regulation for UP's Degree Courses which defines a credit as corresponding to 27 hours of student work.***

**9.3. Forma como os docentes foram consultados sobre a metodologia de cálculo do número de créditos ECTS das unidades curriculares:**
***Após definição do plano curricular contactaram-se os docentes nas áreas de especialidade para avaliar do interesse em lecionar as diferentes UCs do plano. Após realização desse levantamento, e após definição dos pontos principais do conteúdo programático realizaram-se reuniões com os diferentes docentes para avaliar a carga de trabalho e definir o número de créditos ECTS das UCs. Sempre que possível, esta avaliação também teve em consideração o esforço de unidades algo similares oferecidas noutros ciclos de estudo.***

**9.3. Process used to consult the teaching staff about the methodology for calculating the number of ECTS credits of the curricular units:**
***In the definition of the syllabus, contact was made with teachers in the specific domain areas to assess their interest in teaching the different modules planned. After completion of this survey, and after defining the main points of the syllabus, meetings were held with different teachers to evaluate the workload and set the number of ECTS credits for each mosule. Whenever possible, this assessment also took into account the effort similar units offered in other courses involved.***

## 10. Comparação com ciclos de estudos de referência no espaço europeu

**10.1. Exemplos de ciclos de estudos existentes em instituições de referência do Espaço Europeu de Ensino Superior com duração e estrutura semelhantes à proposta:**

***- Masters in Software Engineering (12 meses, tempo integral) -- University of Glasgow, Scotland***
***Inclui aulas, seminários e tutoriais para os estudantes terem a oportunidade de participar em laboratórios, projetos e trabalhos em equipa***
***- Masters in Software Engineering -- University of Liverpool, UK***
***Curso Online***
***- European Masters in Software Engineering (24 meses, tempo integral) -- Technical University of Kaiserlautern, Germany***
***As UCs são orientadas a projeto, oferecidas através de estágios com a indústria e projetos na universidade***
***- Master of Science Computer Science (Software Technology track) (24 meses, tempo integral)-- Delft University of Technology, Netherlands***
***1º ano dedicado a estudo teórico, trabalhos práticos e de laboratório. 2º ano dedicado à especialidade e ao projeto de graduação***
***- Masters in Software Technology (12 meses, tempo integral) -- Dublin Institute of Technology, IrelandPlano de estudos de um ano curricular, com estágio remunerado de 3 meses na Ericsson.***

**10.1. Examples of study programmes with similar duration and structure offered by reference institutions of the European Higher Education Area:**

***Masters in Software Engineering (12 months full-time) -- University of Glasgow, Scotland***
***-The MSc includes lectures, seminars and tutorials and allows students the opportunity to take part in lab, project and team work.***
***Masters in Software Engineering (Online) -- University of Liverpool, UK***
***-Online course***
***European Masters in Software Engineering (two years) -- Technical University of Kaiserlautern, Germany***
***-The courses are project-oriented, which are offered through a wide range of industry internships and university projects for the students.***
***Masters in Software Technology -- Delft University of Technology, Netherlands***
***-The first year comprises theoretical study, assignments and laboratory work. The second year is largely devoted to specialisation and the graduation project.***
***Masters in Software Technology -- Dublin Institute of Technology, Ireland***
***-A one year, full-time MSc with a three month paid internship at Ericsson.***

**10.2. Comparação com objetivos de aprendizagem de ciclos de estudos análogos existentes em instituições de referência do Espaço Europeu de Ensino Superior:**

***No que se refere à estrutura curricular, o MESW é similar aos mestrados de referência no espaço Europeu (e também Internacional, nomeadamente os EUA). Os mestrados existentes no Espaço Europeu são contudo menos abrangentes do que o proposto. Os mestrados analisados, que apresentam elevadas taxas de procura, apresentam uma especialização das áreas da Engenharia de Software, focando-se apenas nos temas para o qual existem investigadores de renome internacional nas Universidades. O MESW, tirando partido do corpo docente qualificado e especializado da Universidade do Porto, apresenta um currículo mais abrangente abordando praticamente todos os tópicos relevantes na área de Engenharia de Software. Uma vantagem importante do MESW é a aproximação da incubadora da U.Porto, sendo que pode ser utilizado pelos estudantes para potenciar ideias de negócio relacionadas com a Engenharia de Software (e.g., ideias que surgirem do trabalho a realizar nas UCs).***

**10.2. Comparison with the intended learning outcomes of similar study programmes offered by reference institutions of the European Higher Education Area:**

***The MESW syllabus is very similar to other highly reputable masters degrees in the EU (as well as globally, namely in the USA). The existent masters in the Euro zone are, however, less comprehensive than the proposed one. The masters we have compared MESW to, which have high rates of demand, specialise in certain areas of Software Engineering, focusing only on topics for which they have internationally recognised academics. MESW, taking advantage of the qualified and specialised faculty of the U. Porto, has a more comprehensive syllabus addressing most areas of Software Engineering. Yet another advantage for MESW is the fact that it can leverage U. Porto's incubator to boost business ideas related to Software Engineering (e.g. ideas may be worked out during the courses).***

# 11. Estágios e/ou Formação em Serviço

## 11.1. e 11.2 Locais de estágio e/ou formação em serviço (quando aplicável)

**Mapa VII - Protocolos de Cooperação**

**Mapa VII - Protocolos de Cooperação**

**11.1.1. Entidade onde os estudantes completam a sua formação:**

***<sem resposta>***

**11.1.2. Protocolo (PDF, máx. 150kB):**
***<sem resposta>***

**Mapa VIII. Plano de distribuição dos estudantes**

**11.2. Mapa VIII. Plano de distribuição dos estudantes pelos locais de estágio e/ou formação em serviço demonstrando a adequação dos recursos disponíveis.(PDF, máx. 100kB).**
***<sem resposta>***

### 11.3. Recursos próprios da Instituição para acompanhamento efectivo dos seus estudantes nos estágios e/ou formação em serviço.

**11.3. Recursos próprios da Instituição para o acompanhamento efectivo dos seus estudantes nos estágios e/ou formação em serviço:**
***<sem resposta>***

**11.3. Resources of the Institution to effectively follow its students during the in-service training periods:**
***<no answer>***

### 11.4. Orientadores cooperantes

**Mapa IX. Normas para a avaliação e selecção dos elementos das instituições de estágio e/ou formação em serviço responsáveis por acompanhar os estudantes**

**11.4.1 Mapa IX. Mecanismos de avaliação e selecção dos orientadores cooperantes de estágio e/ou formação em serviço, negociados entre a Instituição de ensino superior e as instituições de estágio e/ou formação em serviço (PDF, máx. 100kB):**
***<sem resposta>***

**Mapa X. Orientadores cooperantes de estágio e/ou formação em serviço (obrigatório para ciclo de estudos de formação de professores)**

**11.4.2. Mapa X. Orientadores cooperantes de estágio e/ou formação em serviço (obrigatório para ciclo de estudos de formação de professores) / External supervisors responsible for following the students' activities (mandatory for teacher training study programmes)**

| Nome / Name | Instituição ou estabelecimento a que pertence / Institution | Categoria Profissional / Professional Title | Habilitação Profissional (1)/ Professional qualifications (1) | Nº de anos de serviço / Nº of working years |
|---|---|---|---|---|

***<sem resposta>***

## 12. Análise SWOT do ciclo de estudos

**12.1. Pontos fortes:**
- ***Candidatura prevista ao programa Erasmus (e similares) para incentivar à mobilidade.***
- ***Parcerias internacionais previstas.***
- ***Prémios de excelência e de incentivo pedagógico e científico atribuídos a docentes do MESW.***
- ***Instituição e docentes com experiência na utilização de métodos pedagógicos inovadores.***
- ***Docentes ligados a Institutos, Centros e Laboratórios de investigação de qualidade reconhecida.***
- ***Acesso a institutos onde os estudantes podem desenvolver competências em I&D.***
- ***Relacionamento dos docentes com empresas da área das TIC e inovação.***
- ***Caráter empreendedor dos docentes, responsável pela criação de start-ups de base tecnológica.***
- ***Acesso a empresas de base tecnológica onde os estudantes podem desenvolver investigação***

***•A UP dispõe de uma plataforma de e-learning de qualidade reconhecida, o Moodle, utilizada pela maioria das UCs. Têm ocorrido várias experiências pela iniciativa Laboratórios Remotos e Virtuais.***
***•Importância conferida às respostas dos inquéritos aos estudantes.***

**12.1. Strengths:**
***•Expected application to the ERASMUS programme (and similar) to promote mobility.***
***•Expected international partnerships.***
***•Pedagogical excellence and incentive awards given to MESW faculty members.***
***•Experience in applying innovative pedagogical methods by both faculty and institution.***
***•Faculty members linked to prestigious, quality-recognised R&D Institutes and Centres.***
***•Student access to R&D facilities for skill improvement.***
***•Faculty members' active relationships with innovation and ICT companies.***
***•Faculty members' entrepreneurial profile, responsible for launching technological-based startups.***
***•Access to technological-based companies where students may conduct research.***
***•UP provides a popular e-learning platform (Moodle) broadly used by most modules. Several other experiments have been conducted through the "Virtual and Remote Laboratories" initiative.***
***•Student survey results duly noted and valued.***

**12.2. Pontos fracos:**
***Na proposta de um novo ciclo de estudos é difícil identificar lacunas embora elas possam surgir no decorrer do mesmo. De qualquer forma, algo que podemos apontar nesta fase diz respeito:***
***• Ao facto do desempenho pedagógico dos docentes ser levado em conta de forma insuficiente na sua avaliação global.***
***• À falta de uma avaliação sistemática da eficácia na formação contínua de professores e do impacto dessa formação no seu desempenho.***
***• À relativamente baixa taxa de respostas aos inquéritos pedagógicos: na FEUP a taxa varia entre20% e 40%.***

**12.2. Weaknesses:**
***In a new course proposal, it is difficult to identify gaps even though they may arise as it develops. However, we are able to indicate the following at this stage:***
***•Teacher pedagogical performance insufficiently accounted for in the overall assessment.***
***•There is the lack of a systematic effectiveness assessment process in teacher training and impact measurement of such training on their performance.***
***•There is a relatively low response rate to educational surveys: at FEUP, this rate varies between 20% and 40%.***

**12.3. Oportunidades:**
***•Aumento dos índices da internacionalização, tirando partido de possíveis novos programas de intercâmbio de estudantes e docentes com outros países***
***•Aumento de intercâmbio internacional de estudantes, bidirecional, ao abrigo do programa Erasmus e outros.***
***•Existência de entidades (polo tecnológico UPTEC, UPIN) com as quais os estudantes podem interagir com a consequente promoção do empreendedorismo.***
***•Criação de um ciclo de estudos que concilia formação sólida em engenharia de software com investigação e, em simultâneo, uma elevada adequação às necessidades do mercado de trabalho.***
***•Ciclo de estudos envolvendo uma articulação com institutos de investigação e empresas de tecnologias de informação, o que potenciará um aumento da atração de estudantes ao nível do 2º ciclo***
***•O setor das TIC tem atualmente uma elevada empregabilidade mas com uma tendência crescente para a procura de recursos com formação avançada diferenciadora, o que contribui para aumentar o nº e qualidade dos candidatos.***

**12.3. Opportunities:**
***•Increasing internationalisation rates, taking advantage of potential new students and faculty exchange programmes with other countries.***
***•Increasing international, bilateral exchange of students under the ERASMUS programme and others.***
***•Students may interact with existing platforms (UPTEC tech node, UPIN) to promote entrepreneurship.***
***•Creation of a course that reconciles solid academic qualification in Software Engineering with research, while simultaneously providing an adequate response to the needs of the job market.***
***•Course which articulates with ICT companies and R&D institutes, giving rise to a greater adherence of 2nd cycle students.***
***•The ICT sector boasts a high employability rate, with a tendency to grow into a procurement of highly skilled, differentiating professionals, thus contributing to an increased number of applicants.***

**12.4. Constrangimentos:**
***• Novo ECDU e regras de avaliação de desempenho podem condicionar a disponibilidade de docentes para tarefas de ensino e de gestão universitária.***
***• Evolução demográfica com redução de estudantes a ingressar no ensino superior.***

***• Redução do financiamento para o Ensino Superior Público poderá ter consequências negativas nos processos atualmente existentes, pela insuficiência de recursos para os executar.***
***• A crise económica leva a que muitos candidatos prefiram optar por procurar um emprego no final dos 3 primeiros anos (no caso, o equivalente a um 1º ciclo de Bolonha) em vez de optar por continuar e obter o grau de Mestre.***
***• Elevado grau de empregabilidade para estudantes da área das tecnologias da informação.***

**12.4. Threats:**

***•New ECDU (University Teacher Career Statutes) and performance evaluation regulation may affect faculty availability for teaching and management tasks.***
***•Demographics evolution into a lower student application rate to Higher Education courses.***
***•Public Higher Education funding restrictions may negatively affect the current course proposals, rendering them unviable due to insufficient resources.***
***•The global economic crisis forces many applicants to opt for employment at the end of the 1st study cycle (3 years), instead of proceeding to a Master's Degree (2nd cycle).***
***•High employability rate for students in ICT areas.***

**12.5. CONCLUSÕES:**

***A Engenharia de Software surge na década de 1980 após a "crise de software" dos anos de 1960, 1970 e 1980 onde se identificaram muitos problemas no desenvolvimento de software. Muitos projetos de software ultrapassaram o orçamento e cronograma. Alguns projetos causaram danos à propriedade e/ou perda de vidas.***
***Os engenheiros de software são profissionais essenciais para a melhoria da qualidade nos sistemas de software. Eles são responsáveis pelo ciclo de vida completo de um produto de software novo ou modificado, desde a pesquisa e conceção até à implementação, formação e suporte.***
***Dada a necessidade de Engenheiros de Software especializados e a não suficiente oferta formativa, cremos que as universidades têm a obrigação de atuar como catalisadores de conhecimento na área, e desse modo contribuir de forma efetiva para a qualidade dos sistemas de software.***
***Este documento apresenta uma proposta para um Mestrado em Engenharia de Software (MESW) de segundo ciclo que tem por objetivo oferecer formação científica e profissional avançada em Engenharia de Software promovendo a inovação e incremento da competitividade.***
***O MESW define objetivos de aprendizagem, operacionalizados pelas suas unidades curriculares e verificados através da análise do percurso académico dos estudantes, tendo como referência o perfil necessário para profissionais capazes de, dada uma necessidade, aplicar uma abordagem sistemática, disciplinada e quantificável conducente ao desenvolvimento, operação e manutenção de um sistema de software que satisfaça os requisitos funcionais e os atributos de qualidade.***
***O MESW pretende ser uma oferta formativa internacional com base no estabelecimento de parcerias internacionais e numa candidatura ao programa Erasmus+.***
***Este ciclo de estudos pode também influenciar de forma profunda e positiva a maneira como as Universidades Portuguesas, e a Universidade do Porto em particular, constroem a sua oferta de mestrados profissionais, pensados para formar líderes técnicos para a indústria, altamente qualificados.***

**12.5. CONCLUSIONS:**

***The concept of Software Engineering arose in the 1980s after the "software crisis" of the 1960s, '70s and '80s which identified many problems in software development. Many software projects exceeded both budget and schedule. Some projects caused property damage and/or loss of life.***
***Software engineers are vital professionals for improving the quality of software systems. They are responsible for the complete life cycle of a new, or modified, software product, from conception and design to implementation, training and support.***
***Given the need for specialised software engineers and the lack of available training, we believe that universities have an obligation to act as knowledge catalysts in the SE area, thereby contributing effectively to the quality of software systems.***
***This document presents a proposal for a Master Degree (second cycle) in Software Engineering (MESW) that aims at providing advanced scientific and professional training in Software Engineering, promoting innovation and increased competitiveness.***
***MESW learning objectives are upheld by its modules and verified through analysis of the academic path taken by its students. Its definition takes, as a reference, the profile required for professionals who are capable of applying a systematic, disciplined and quantifiable approach, leading to the development, operation and maintenance of a software system that meets the functional requirements and quality attributes.***
***The MESW intends to be an international training offer, built on international partnerships and an application to the Erasmus + programme.***
***This course can also deeply and positively influence the way the Portuguese Universities, and the University of Porto in particular, build their range of professional masters degrees designed to produce technical, highly skilled, leaders for the industry.***