

Plano Analítico: Engenharia de Software

1. Identificação da Unidade Curricular

- **Instituição:** Instituto Superior Politécnico de Ciências e Tecnologia (INSUTEC)
- **Curso:** Engenharia de Informática e Sistemas de Informação (EISI)
- **Classificação:** Disciplina Específica (Nuclear)
- **Ano:** 4º | **Semestre:** 2º (8º Semestre)
- **Créditos:** 8.0 UC
- **Carga Horária Total:** 120 Horas (90h de Contacto | 30h de Trabalho Complementar)

2. Apresentação e Justificação

A disciplina de Engenharia de Software no último ano funciona como o ponto de convergência de todo o conhecimento técnico adquirido no curso. O foco transita do "como programar" para o "como construir sistemas de software complexos, fiáveis e sustentáveis". Aborda a gestão de configuração, a garantia de qualidade (QA) e a manutenção de sistemas legados. É vital para preparar o estudante para o mercado de trabalho e para a execução do seu Trabalho de Fim de Curso (TFC), conforme o **Decreto Presidencial 193/18**.

3. Competências a Desenvolver (Decreto 193/18)

3.1 Competências Instrumentais (Saber)

- Compreender os modelos de maturidade de processos (CMMI e MPS.BR).
- Conhecer as métricas de software para estimativa de custos e prazos (Pontos de Função).
- Entender os princípios de DevOps e Integração Contínua (CI/CD).

3.2 Competências Técnicas e Operacionais (Saber Fazer)

- **Gestão de Configuração:** Dominar o controlo de versões e a gestão de *releases* de software.
- **Garantia de Qualidade:** Implementar planos de teste automatizados e auditorias de código.
- **Manutenção:** Aplicar técnicas de refatoração e engenharia reversa em sistemas existentes.

3.3 Competências Atitudinais (Saber Ser/Estar)

- Demonstrar liderança e capacidade de trabalho em equipas multidisciplinares.
- Comprometer-se com o cumprimento de prazos e com a entrega de valor ao cliente final.

4. Conteúdo Temático (Estrutura de 120 Horas)

1. **Processos de Software Avançados:** Metodologias Ágeis (Scrum, Kanban) vs. Modelos Híbridos.
2. **Gestão de Configuração de Software (SCM):** Repositórios, ramificação (*branching*) e fusão (*merging*) de código.

3. **Métricas e Estimativas:** Medição de tamanho, esforço e produtividade; análise de pontos de função.
4. **Qualidade de Software (QA/QC):** Normas ISO/IEC 25010, testes de carga, segurança e usabilidade.
5. **Manutenção e Evolução:** Lei de Lehman, refatoração de código e gestão da dívida técnica.
6. **Tópicos Emergentes:** Microserviços, *Serverless* e o impacto da IA na geração de código.

5. Regime de Avaliação (Disciplina Específica)

- **Avaliação Contínua (40%):**
 - 1ª Frequência (Processos e Métricas): 13%
 - 2ª Frequência (Qualidade e Manutenção): 14%
 - **Projeto Integrador:** Simulação de um ciclo completo de desenvolvimento ágil: 13%
- **Exame Normal (60%):** Prova global teórica com foco em gestão de projetos de software.

6. Referências Bibliográficas (APA 7ª Ed.)

- Sommerville, I. (2019). *Engenharia de software* (10ª ed.). Pearson.
- Pressman, R. S., & Maxim, B. R. (2021). *Engenharia de software: Uma abordagem profissional* (9ª ed.). McGraw-Hill.
- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org.
- Martin, R. C. (2008). *Clean Code: A handbook of agile software craftsmanship*. Prentice Hall.