



**UNIVERSIDADE FEDERAL DE VIÇOSA**  
**PRÓ REITORIA DE ENSINO**  
**DIRETORIA DE REGISTRO ESCOLAR**

**Programa Analítico de Disciplina**

**INF221 Engenharia de Software I**

Departamento de Informática - Centro de Ciências Exatas e Tecnológicas

Número de créditos: 4		<u>Teóricas</u>	<u>Práticas</u>	<u>Total</u>
Duração em semanas: 15	Carga horária semanal	4	0	4
Períodos - oferecimento: I	Carga horária total	60	0	60

Pré-requisitos (Pré ou co-requisitos)\*

INF213

**Ementa**

Contexto para projeto de programas. Qualidade de software no nível pessoal. Requisitos de pequenos sistemas e programas. Projeto de interfaces com o usuário. Projeto preliminar. Projeto detalhado e implementação. Teste de programas e sistemas. Tópicos complementares.

**Oferecimento aos Cursos**

<b>Curso</b>	<b>Modalidade</b>	<b>Período</b>
Ciência da Computação	Obrigatória	5



**UNIVERSIDADE FEDERAL DE VIÇOSA**  
**PRÓ REITORIA DE ENSINO**  
**DIRETORIA DE REGISTRO ESCOLAR**

**INF221 Engenharia de Software I**

<b>Seq</b>	<b>Aulas Teóricas</b>	<b>Horas/Aula</b>
1	Contexto para projeto de programas  1.1. A crise do software e suas possíveis soluções 1.2. Problemas com prazo, planejamento e custos 1.3. Qualidade de software 1.4. Processos de desenvolvimento e ciclos de vida 1.5. Métodos de projeto de software 1.6. Qualidade do ponto de vista pessoal 1.7. Qualidade de processos 1.8. Ciclo de desenvolvimento em cascata 1.9. Ciclo de prototipação 1.10. Orientação a objetos, reusabilidade e uso de componentes	4
2	Qualidade de software no nível pessoal  2.1. Introdução ao Processo Pessoal de Software (PSP) 2.2. Histórico, qualidade de produto e de processo, níveis de qualidade, relação com o Modelo de Capacitação de Maturidade (CMM) 2.3. Formulários, processo de adoção 2.4. Estabelecimento de padrões 2.5. Análise e melhoria do processo pessoal de desenvolvimento 2.6. Adoção de padrões de teste e construção de casos / de teste 2.7. Adoção de ciclo pessoal de desenvolvimento	8
3	Requisitos de pequenos sistemas e programas  3.1. Requisitos funcionais e não funcionais 3.2. Análise estruturada de sistemas e métodos baseados em fluxo de dados 3.3. Dicionário de dados 3.4. Especificação de atividades com pseudo-código 3.5. Revisões formais 3.6. Especificação do banco de dados e sua relação com as aplicações	8
4	Projeto de interfaces com o usuário  4.1. Prototipação da interface para entendimento de requisitos 4.2. Fatores que afetam a qualidade de interfaces 4.3. Classes de usuários 4.4. Classes de interfaces e sua relação com classe de usuários 4.5. Principais modelos para construção de interfaces: formulários, manipulação direta, seleção por menu, linguagem de comando 4.6. Gerenciamento de erros e diálogos com o usuário 4.7. Documentação e treinamento do usuário	8



**UNIVERSIDADE FEDERAL DE VIÇOSA**  
**PRÓ REITORIA DE ENSINO**  
**DIRETORIA DE REGISTRO ESCOLAR**

5	Projeto preliminar  5.1. Modularidade 5.2. Projeto da estrutura do software 5.3. Análise de transformações e análise de transações 5.4. Derivação da estrutura modular 5.5. Projeto das estruturas de dados 5.6. Relação entre projeto da aplicação e o projeto do banco de dados do sistema 5.7. Documento de especificação preliminar 5.8. Revisões formais e caminhamentos estruturados	12
6	Projeto detalhado e implementação  6.1. Projeto e implementação dos algoritmos principais 6.2. Critérios para escolha da linguagem de programação 6.3. Fatores de risco associados à escolha da linguagem 6.4. O processo de codificação e a integração dos módulos 6.5. Qualidade de programas e adoção de padrões 6.6. Revisões e caminhamentos estruturados 6.7. Documentação	10
7	Teste de programas e sistemas  7.1. Níveis de teste 7.2. Princípios de teste 7.3. Projeto de casos de teste 7.4. Teste de módulo e de sistema 7.5. Estratégias de teste de sistema 7.6. Documentação	6
8	Tópicos complementares  8.1. A visão da Engenharia de Software 8.2. Tendências 8.3. Programação e projeto orientados por objetos 8.4. Ética profissional e responsabilidade social	4



**UNIVERSIDADE FEDERAL DE VIÇOSA  
PRÓ REITORIA DE ENSINO  
DIRETORIA DE REGISTRO ESCOLAR**

**INF221 Engenharia de Software I**

**Referências Bibliográficas**

**Bibliografia Básica:**

---

**Bibliografia Complementar:**

- 1 - BENTLEY, J. Programming Pearls. Reading, Massachussets: Addison-Wesley, 1999. [Exemplares disponíveis: Não informado.]
- 2 - HUMPHREY, W.S. Introduction to the personal software process. Reading, Massachussets: Addison-Wesley, 1996. [Exemplares disponíveis: Não informado.]
- 3 - KERNIGHAN, B.W.; PIKE, R. The practice of programming. Reading, Massachussets: Addison-Wesley, 1999. [Exemplares disponíveis: Não informado.]
- 4 - MYNATT, B.T. Software engineering with student project guidance. Englewood Cliffs, NJ: Prentice-Hall, 1990. [Exemplares disponíveis: Não informado.]
- 5 - PAULA FILHO, W.P. Engenharia de software: fundamentos, métodos e padrões. Rio de Janeiro, RJ: LTC, 2001. [Exemplares disponíveis: Não informado.]
- 6 - PRESSMAN, R. Software Engineering: a practitioner's approach. New York, NY, McGraw-Hill, 2000. [Exemplares disponíveis: Não informado.]
- 7 - RUMBAUGH, J. et al. Modelagem e projeto baseados em objetos. Rio de Janeiro, RJ, Ed. Campus, 1994. [Exemplares disponíveis: Não informado.]
- 8 - SHNEIDERMAN, B. Designing the user interface: strategies for effective human-computer interaction. Reading, Massachussets, Addison-Wesley, 1997 [Exemplares disponíveis: Não informado.]
- 9 - SOMMERVILLE, I. Software Engineering. Reading, Massachussets, Addison-Wesley, 2001. [Exemplares disponíveis: Não informado.]
- 10 - STAA, A.V. Programação Modular. Rio de Janeiro, RJ, Ed. Campus, 2000. [Exemplares disponíveis: Não informado.]
- 11 - YOURDON, E. Análise estruturada moderna. Rio de Janeiro, RJ, Ed. Campus, 1990. [Exemplares disponíveis: Não informado.]