

**Plano de Disciplina:** Design de Software

**Carga Horária:** 80 horas

### **Ementa**

- Arquitetura de software e modelagem de sistemas de informação. Atividades de modelagem em um ciclo de vida e no processo de software. Técnicas de levantamento de dados, especificação de processos, requisitos funcionais e não funcionais de software e regras de negócio. Declaração do escopo do projeto de software. Abordagens de modelagem Orientada a Objetos. Projetos de software Orientados a Objetos. UML aplicado à modelagem O.O. Elaboração de cronograma do projeto de software para apoio à gestão do tempo, monitoramento e controle de tarefas. Gerenciamento de configuração e mudanças.
- Modelagem de sistemas assistida por computador (CASE).

### **Objetivos**

- Fazer levantamento de requisitos
- Usar ferramentas CASE
- Modelar aplicações orientadas a objetos em UML a partir de uma especificação de requisitos.
- Representar corretamente funcionalidades do sistema utilizando UML.
- Documentar sistemas utilizando a UML.
- Utilizar ferramentas para Modelagem OO.

### **Conteúdos**

- 1) Conceitos básicos de orientação a objetos
    - a) Objetos
    - b) Classes
    - c) Atributos
    - d) Métodos
    - e) Estado, comportamento e identidade de um objeto
    - f) Encapsulamento
    - g) Dependência entre classes
    - h) Associações: multiplicidade e navegabilidade
    - i) Herança
    - j) Polimorfismo
    - k) Troca de mensagens entre objetos
  - 2) Introdução ao UML
    - a) Diagramas Estruturais
    - b) Diagrama de objetos
    - c) Diagrama de classes
-

- d) Diagrama de componentes
  - e) Diagrama de instalação
  - f) Diagrama de pacotes
  - g) Diagrama de estrutura
  - h) Diagramas Comportamentais
  - i) Diagrama de Caso de Uso
  - j) Diagrama de Estados
  - k) Diagrama de atividade
  - l) Diagramas de Interação
  - m) Diagrama de sequência
  - n) Diagrama de Interatividade
  - o) Diagrama de colaboração
  - p) Diagrama de tempo
- 3) Mapeamento Objeto x Relacional.
- 4) Ferramentas para Modelagem OO.

### **Metodologia de Ensino**

- Apresentação de modelos de Aplicações
- Aulas expositivas e dialógicas
- Leituras Complementares
- Atividades de laboratórios virtuais
- Resolução de exercícios e problemas
- Pesquisas e seminários
- Jogos e experiências práticas de aplicação
- Debates
- Utilização de aplicativos, filmes, publicações, divulgação científica da web, livros

### **Bibliografia Básica:**

GALLOTTI, G. M. A. (Org.). Arquitetura de software. Pearson: 2017.

SOMMERVILLE, I. Engenharia de software. 10.ed. Pearson: 2019

PFLEEGER, S. L. Engenharia de software: teoria e prática. 2.ed. Pearson: 2003

### **Bibliografia Complementar:**

GALLOTTI, G. M. A. Qualidade de software. Pearson: 2015

BRAGA, P. H. Teste de software. Pearson: 2016

MEDEIROS, E. Desenvolvendo software com UML 2.0 definitivo. Pearson: 2004

ASCENCIO, A. F. G.; CAMPOS, E. A. V. de. Fundamentos da programação de computadores. 3.ed. Pearson: 2012

---