



## **PLANO DE ENSINO**

### **1. IDENTIFICAÇÃO**

Campus: Avançado Quedas do Iguaçu

Eixo tecnológico: Informação e Comunicação

Curso: Técnico em Informática Integrado ao Ensino Médio

Componente curricular: Análise e Projeto de Sistemas

Docente: Rafael Gil Ferques

Carga horária: 67 horas - 80 h/a

Turno: Matutino

Número de aulas na semana: 02 aulas semanais

Período letivo: 2019

Turma (s): 4º Ano

Coordenador do curso: Odair Moreira de Souza

### **2. EMENTA**

A Análise e Projeto de Sistemas e a Engenharia de Software: Contextualização da Análise e Projeto de Sistemas dentro da Engenharia de Software. O papel dos Sistemas de Informação para os diversos segmentos da sociedade. Vantagens e desvantagens do desenvolvimento de software e da utilização de softwares integrados de gestão empresarial. Evolução da arquitetura de software; Análise Orientada a Objetos: Classes e Objetos, Mensagens, Encapsulamento, Polimorfismo, Herança; Metodologias de desenvolvimento de software.

### **3. OBJETIVOS**

#### **3.1 Objetivo geral**

Compreender o processo de desenvolvimento de software, as atividades técnicas e as interações envolvidas neste processo, desde a análise preliminar até o desenvolvimento do software.

#### **3.2 Objetivos específicos**

- Identificar os principais conceitos para análise de sistemas, desde a fase problema até a modelagem de dados.
- Compreender o funcionamento e importância da fase de levantamento de requisitos.
- Compreender os principais pontos da modelagem de dados.



- Adquirir conhecimentos sobre conceito, técnicas e métodos para análise, projeto e implementação de sistemas computacionais.
- Elaborar instrumentos de coleta de dados direcionados ao levantamento de requisitos de software.
- Elaborar um projeto lógico a fim de implementar um software a partir da documentação resultante.

## **4. CONTEÚDO PROGRAMÁTICO**

### **4.1. Introdução**

### **4.2. Fase de planejamento**

#### 4.2.1. Introdução

#### 4.2.2. O analista de sistemas

##### 4.2.2.1. Habilidades do Analista de sistemas

##### 4.2.2.2. As funções do Analista de sistemas

#### 4.2.3. Ciclo de vida de desenvolvimento de sistemas

##### 4.2.3.1. Planejamento

##### 4.2.3.2. Análise

##### 4.2.3.3. Projeto

##### 4.2.3.4. Implementação

#### 4.2.4. Identificação e iniciação do projeto

#### 4.2.5. Análise de viabilidade

##### 4.2.5.1. Viabilidade técnica

##### 4.2.5.2. Viabilidade econômica

##### 4.2.5.3. Viabilidade organizacional

#### 4.2.6. Seleção e gerenciamento de projeto

##### 4.2.6.1. Seleção de projeto

##### 4.2.6.2. Criando plano de projeto

##### 4.2.6.3. Gerenciando projeto

### **4.3. Determinação dos requisitos**

#### 4.3.1. Introdução

#### 4.3.2. Determinação de requisitos

##### 4.3.2.1. Definição de requisito

##### 4.3.2.2. O processo de determinação

##### 4.3.2.3. Desenvolvimento de conjunto de aplicações

##### 4.3.2.4. Análise de documentos

#### 4.3.3. Estratégias de análise de requisitos

##### 4.3.3.1. Análise do problema

##### 4.3.3.2. Análise da causa principal

##### 4.3.3.3. Análise de duração

##### 4.3.3.4. Custo baseado em atividades

##### 4.3.3.5. Análise de tecnologia



4.3.3.6. Comparação de resultados

**4.4. Análise de casos de uso**

4.4.1. Introdução

4.4.2. Caso de usos

4.4.2.1. Elementos de um caso de uso

4.4.2.2. Formatos alternativos de casos de uso

4.4.2.3. Casos de uso e os requisitos funcionais

4.4.2.4. Casos de uso e testes

4.4.2.5. Construção de casos de uso

4.4.2.6. Identificação dos casos de uso principais

4.4.2.7. Elaboração de casos de uso

4.4.3. Modelagem de dados

4.4.3.1. Importância da modelagem

4.4.3.2. Princípios da modelagem

4.4.3.3. Modelagem orientada a objetos

**4.5. UML – Unified modeling language**

4.5.1. Visão geral da UML

4.5.2. Modelo conceitual da UML

4.5.2.1. Bloco de construção

4.5.2.2. Itens

4.5.2.3. Regras

4.5.2.4. Mecanismos básicos

4.5.3. Arquitetura

4.5.4. Ciclo de vida do desenvolvimento de software

**4.6. Modelagem estrutural básica**

4.6.1. Classes

4.6.1.1. Conceitos gerais

4.6.1.2. Termos e conceitos

4.6.1.3. Técnicas básicas de modelagem

4.6.2. Relacionamentos

4.6.2.1. Conceitos gerais

4.6.2.2. Dependência

4.6.2.3. Generalização

4.6.2.4. Associação

4.6.3. Mecanismos

4.6.3.1. Notas

4.6.3.2. Estereótipos, valores de atributos e restrições

4.6.3.3. Comentários

4.6.3.4. Valores atribuídos

4.6.4. Diagramas

4.6.4.1. Diagramas, visões e modelos

**4.6.5. Diagrama de classes**



- 4.6.5.1. Conceitos gerais
- 4.6.5.2. Classes
- 4.6.5.3. Associações
- 4.6.5.4. Generalizações e especializações

#### **4.6.6. Diagrama de objetos**

- 4.6.6.1. Conceitos gerais
- 4.6.6.2. Classes
- 4.6.6.3. Associações
- 4.6.6.4. Generalizações e especializações
- 4.6.6.5. Construção do modelo de classes

#### **4.6.7. Diagrama de entidade relacionamento – DER**

- 4.6.7.1. Conceitos gerais
- 4.6.7.2. Relacionamentos e Cardinalidade
- 4.6.7.3. Identificação de relacionamentos

#### **4.6.8. Modelo entidade relacionamento – MER**

- 4.6.8.1. Conceitos gerais
- 4.6.8.2. Entidades
- 4.6.8.3. Relacionamentos
- 4.6.8.4. Atributos
- 4.6.8.5. Dicionário de dados

#### **4.6.9. Diagrama de Atividades**

- 4.6.9.1. Fluxo de controles
- 4.6.9.2. Diagrama de atividades no fluxo interativo

#### **4.6.10. Diagrama de Interação**

- 4.6.10.1. Elementos da modelagem de interações
  - 4.6.10.1.1. Mensagens, atores, objetos e classes

#### **4.6.11. Diagrama de transição de estados**

- 4.6.11.1. Conceitos gerais
- 4.6.11.2. Identificação dos elementos de um diagrama de estado
- 4.6.11.3. Construção
- 4.6.11.4. Modelagem de estados no processo de desenvolvimento

#### **4.6.12. Diagrama de Sequência**

- 4.6.12.1. Linhas de vida
- 4.6.12.2. Mensagens
- 4.6.12.3. Ocorrência de execução
- 4.6.12.4. Criação e exclusão de objetos

#### **4.6.13. Diagrama de Colaboração**

- 4.6.13.1. Conceitos gerais
- 4.6.13.2. Relacionamento de classes

### **4.7. Outros Diagramas**

## **5. METODOLOGIA DE ENSINO E AVALIAÇÃO**



Aulas expositivas seguidas de aulas práticas com aplicação dos conceitos apresentados em sala de aula e em laboratório. Aplicação de listas de exercício para auxiliar na fixação de conteúdo. Resolução das listas em laboratório de informática, priorizando a resolução de dúvidas oriundas das tentativas de resolução das listas propostas. Análise de estudos de casos apresentados em seminários e oficinas. Discussão de atividades em grupo; debates em sala de aula e em laboratório de informática e constante interação entre o docente e os discentes referente aos assuntos abordados.

Quanto à avaliação, tendo-a caráter permanente e diagnóstico, como parte do processo de ensino e de aprendizagem, esta poderá ser realizada mediante atividades diversas tais como: leitura, compreensão e síntese dos tópicos abordados; provas; abertura de diálogos, debates e exposição oral de conclusões via seminário ou trabalhos em grupo, além do registro escrito por meio de textos ou respostas de questionamentos. A pesquisa e as leituras complementares serão incentivadas, podendo ser apresentadas como forma de avaliação. A participação do aluno ouvindo, questionando, argumentando, assim como sua postura respeitosa e acadêmica, são pontos fundamentais da avaliação. Os resultados serão apresentados a cada trabalho, sendo explicitado o diagnóstico feito pelo docente. De acordo com as normas da Instituição, os alunos receberão os conceitos A, B, C ou D nos períodos determinados pelo IFPR e no final do conteúdo de cada área curricular.

### 5.1 Recursos didáticos

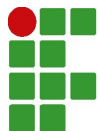
Para o desenvolvimento das aulas serão utilizados, ao longo do componente curricular, os seguintes recursos:

- Quadro branco;
- Livros didáticos;
- Projeto Multimídia;
- Laboratório com programas específicos.

## 6. CRITÉRIOS DE AVALIAÇÃO

A avaliação dos alunos será realizada de acordo com a Resolução Nº 50 de 14 de Julho de 2017 do IFPR, através dos seguintes mecanismos:

- Seminários;
- Trabalhos individuais e/ou em grupos;
- Testes escritos e/ou orais/sinalizados;
- Demonstrações de técnicas em laboratório;
- Autoavaliações.



A cada bimestre, o conceito do aluno será composto por meio de sua participação e aproveitamento nos quesitos descritos anteriormente, tanto de forma somativa quanto formativa.

A definição do número de instrumentos para avaliação também serão norteados pela mesma resolução supracitada, que no caso serão no mínimo dois mecanismos.

## 7. RECUPERAÇÃO

De acordo com a Lei nº 9.394, de 20 de dezembro de 1996, a oferta de estudos de recuperação é obrigatória. Esta será ofertada aos estudantes paralelamente ao período letivo, para os casos de baixo rendimento escolar.

Serão adotados os seguintes mecanismos de recuperação:

- Recuperação Paralela;
- Recuperação Contínua;

Para nortear os mecanismos adotados se utilizará a Resolução Nº 50 de 14 de Julho de 2017 do IFPR, que prevê a forma de tratamento para os mesmos.

## 8. BIBLIOGRAFIA

### 8.1 Bibliografia Básica

1. BEZERRA, E. **Princípios de Análise e Projeto de Sistemas com UML**. 3 ed. Rio de Janeiro: *Elsevier*, 2015, 369 p.
2. BOOCH, G.; RUMBAUGH, J. JACOBSON, I. **UML: guia do usuário**. 2 ed. Rio de Janeiro: *Elsevier*, 2012, 521p.
3. PAULA FILHO, W. P. **Engenharia de Software: fundamentos, métodos e padrões**. 3 ed. Rio de Janeiro: LTC, 2015, 1248 p.
4. PRESSMAN, R. S; MAXIM, B. R. **Engenharia de software: uma abordagem profissional**. 8 ed. Porto Alegre: AMGH, 2016, 940 p.
5. SOMMERVILLE, I. **Engenharia de software**. 9 ed. São Paulo: Pearson Prentice Hall, 2011, 529 p.

### 8.2 Bibliografia Complementar



1. CARDOSO, C. **Uml na Prática: Do Problema ao Sistema**. Rio de Janeiro: Ciência Moderna, 2003, 120 p.
2. ENGHOLM JUNIOR, H. **Engenharia de software na prática**. São Paulo: Novatec, 2010. 438 p.

## 9. OBSERVAÇÕES

Quedas do Iguaçu, 22 de fevereiro de 2019.

---

**Rafael Gil Ferques**  
Docente