

# **ARTIGO CIENTÍFICO: SIMULAÇÃO COMPUTACIONAL DE PROCESSOS DE ENGENHARIA DE SOFTWARE: UMA ABORDAGEM PRÁTICA BASEADA NA UNIDADE 1 DO LIVRO "ANÁLISE E MODELAGEM DE SISTEMAS"**

Autores: Caio Góes

Daniel Queiroz

João Miguel Costa Silva

Anhanguera unidade de Santana

**Resumo:** Este artigo apresenta uma implementação computacional em Python que simula processos de engenharia de software, baseando-se nos fundamentos teóricos apresentados na Unidade 1 do livro "Análise e Modelagem de Sistemas". O estudo demonstra a aplicação prática dos conceitos de processos de software, métricas de qualidade e análise de viabilidade através de um simulador desenvolvido no ambiente Google Colab. A ferramenta permite visualizar e analisar diferentes cenários de desenvolvimento, facilitando a compreensão dos modelos prescritivos e ágeis discutidos na literatura. Os resultados mostram a eficácia da simulação computacional como ferramenta educacional para o ensino de engenharia de software.

Palavras-chave: Engenharia de Software. Processos de Software. Simulação Computacional. Análise de Sistemas. Python.

## **1. INTRODUÇÃO**

A engenharia de software constitui-se como disciplina fundamental para o desenvolvimento sistemático de sistemas computacionais de qualidade. Conforme apresentado na Unidade 1 do livro "Análise e Modelagem de Sistemas" (EDITORA EDUCACIONAL, 2020), a complexidade inerente aos processos de desenvolvimento requer abordagens estruturadas e metodologias bem definidas para garantir o sucesso dos projetos.

A simulação computacional emerge como ferramenta valiosa para demonstrar na prática os conceitos teóricos discutidos na literatura especializada. Pressman (2016) e Sommerville (2011) destacam a importância de modelos e ferramentas para melhor compreensão dos processos de software, particularmente na fase de análise e planejamento.

Este artigo apresenta uma implementação prática dos conceitos da Unidade 1 através de um simulador desenvolvido em Python, abordando especificamente os processos de software, métricas de qualidade e análise de viabilidade discutidos nas seções 1 e 2 do livro de referência.

## **2. OBJETIVO**

Desenvolver e analisar uma ferramenta de simulação computacional que implemente os conceitos de engenharia de software apresentados na Unidade 1 do livro "Análise e Modelagem de Sistemas", especificamente:

- a) Modelar processos de software com suas fases características;
- b) Implementar métricas de qualidade e produtividade;
- c) Desenvolver algoritmo para análise de viabilidade de projetos;
- d) Visualizar resultados através de representações gráficas.

## **3. METODOLOGIA**

A pesquisa caracteriza-se como aplicada, de natureza quantitativa, utilizando simulação computacional como método principal. O desenvolvimento foi realizado em Python no ambiente Google Colab, com as seguintes etapas:

3.1. Modelagem do Processo de Software Implementou-se a classe `ProcessoSoftware` baseada nos modelos prescritivos discutidos por Pressman (2016), contendo as cinco fases fundamentais: requisitos, projeto, implementação, testes e implantação.

3.2. Métricas de Qualidade Foram implementadas métricas baseadas na ISO/IEC 9126, considerando:

- Tempo de desenvolvimento por fase

- Custo planejado versus realizado
- Defeitos identificados por fase
- Taxa de correção de defeitos

3.3. Algoritmo de Análise de Viabilidade Desenvolveu-se função baseada em múltiplos critérios (custo, tempo, complexidade, satisfação do usuário e vida útil restante) para decisão entre atualização de sistema legado ou desenvolvimento de novo sistema.

3.4. Visualização de Dados Utilizou-se as bibliotecas Matplotlib e Seaborn para geração de gráficos comparativos e evolutivos dos processos simulados.

## 4. RESULTADOS E DISCUSSÃO

4.1. Simulação do Processo de Software A classe `ProcessoSoftware` demonstrou eficácia na modelagem das fases de desenvolvimento, permitindo acompanhamento em tempo real do progresso e identificação de defeitos por fase (Figura 1). Os resultados alinham-se com a proposição de Sommerville (2011) sobre a importância do acompanhamento sistemático do processo.

```
projeto = ProcessoSoftware("Sistema de Gestão Acadêmica", 5)
```

```
projeto.iniciar_fase("Requisitos")
```

```
projeto.completar_fase("Requisitos", 2)
```

4.2. Análise de Métricas de Qualidade As visualizações geradas permitiram análise comparativa entre valores planejados e realizados, demonstrando a variação comum em projetos de software discutida por Pressman (2016). A Figura 2 mostra disparidades significativas entre tempo estimado e real na fase de implementação.

4.3. Tomada de Decisão para Sistemas Legados O algoritmo de análise de viabilidade demonstrou eficácia em avaliar múltiplos critérios simultaneamente, apoiando a decisão entre atualização ou desenvolvimento

novo. Os resultados corroboram com Engholm Jr. (2010) sobre a importância de análise multidimensional na Engenharia de Software.

Cenário	Decisão Recomendada	Score Atualização	Score Novo
1	Desenvolver Novo	0,42	0,58
2	Atualizar	0,61	0,39
3	Desenvolver Novo	0,38	0,62

*Tabela 1: Resultados da análise de viabilidade para três cenários*

## 5. CONCLUSÃO

A implementação computacional desenvolvida demonstrou-se eficaz na aplicação prática dos conceitos teóricos apresentados na Unidade 1 do livro "Análise e Modelagem de Sistemas". A ferramenta permite visualização concreta dos processos de software, facilitando a compreensão dos desafios inerentes ao desenvolvimento de sistemas.

Os resultados obtidos alinham-se com a literatura especializada, particularmente no que concerne à importância do acompanhamento de métricas de qualidade e da análise multidimensional para tomada de decisão em projetos de software.

Como trabalhos futuros, sugere-se a expansão do simulador para incluir modelos ágeis de desenvolvimento e integração com ferramentas de gestão de projetos reais.

## REFERÊNCIAS

EDITORA EDUCACIONAL. Análise e Modelagem de Sistemas. Londrina: Editora e Distribuidora Educacional, 2020.

ENGHOLM JR., H. Engenharia de Software na Prática. São Paulo: Novatec, 2010.

PRESSMAN, R. S.; MAXIM, B. R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016.

SOMMERVILLE, I. Engenharia de software. 9. ed. São Paulo: Pearson Prentice Hall, 2011.