



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PLANO DE DISCIPLINA			
IDENTIFICAÇÃO			
CAMPUS: Esperança - PB			
CURSO: Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas			
DISCIPLINA: Gerência de Configuração e Mudanças	CÓDIGO DA DISCIPLINA: 62		
PRÉ-REQUISITO: Desenvolvimento de Aplicações Web II			
UNIDADE CURRICULAR: Obrigatória [x] Optativa [] Eletiva []	SEMESTRE/ANO: 6º		
CARGA HORÁRIA			
TEÓRICA: 15h	PRÁTICA: 18h	EaD¹: 0h	EXTENSÃO: 0h
CARGA HORÁRIA SEMANAL: 2h/a			
CARGA HORÁRIA TOTAL: 33h			
DOCENTE RESPONSÁVEL: Alvaro Magnum Barbosa Neto			

EMENTA

Ciclo de vida de produtos e artefatos (Conceitos de Artefatos e Produtos; Estruturação de Artefatos e Produtos; Noções sobre produção de produtos). Noções de gerenciamento de configurações de artefatos e produtos. Noções sobre controle de mudanças. Principais ferramentas. Noções sobre integração contínua.

OBJETIVOS DA DISCIPLINA/COMPONENTE CURRICULAR

Objetivo Geral:

Capacitar os alunos a compreenderem e aplicarem conceitos, metodologias e ferramentas relacionadas ao ciclo de vida de produtos e artefatos, gestão de configurações e controle de mudanças, promovendo boas práticas e integração contínua no desenvolvimento de sistemas.

Objetivos Específicos:

1. Apresentar os conceitos fundamentais de artefatos e produtos, bem como as técnicas de estruturação e produção associadas.
2. Ensinar noções de gerenciamento de configurações, abordando a organização e rastreabilidade de artefatos e produtos.
3. Introduzir práticas de controle de mudanças, com foco em minimizar riscos e melhorar a qualidade dos projetos.
4. Explorar as principais ferramentas utilizadas no gerenciamento de configurações e mudanças no mercado.
5. Demonstrar os princípios da integração contínua e sua aplicação no contexto do desenvolvimento de software.

CONTEÚDO PROGRAMATICO

Etapa 1: Introdução aos Conceitos de Artefatos e Produtos

- Definições e diferenças entre artefatos e produtos.
- Estruturação e organização de artefatos e produtos.
- Ciclo de vida de produtos e artefatos.
- Noções de produção e controle de qualidade de produtos.

Etapa 2: Gerenciamento de Configurações

- Conceitos e importância do gerenciamento de configurações.

- Técnicas de organização e rastreabilidade de configurações.
- Estruturação de repositórios e controle de versões.
- Práticas e diretrizes para manter a consistência de configurações.

Etapa 3: Controle de Mudanças

- Definição e impacto do controle de mudanças em projetos de software.
- Técnicas para análise de impactos e planejamento de mudanças.
- Processos e fluxos de aprovação de mudanças.
- Diretrizes para minimizar riscos e retrabalhos em mudanças.

Etapa 4: Ferramentas e Integração Contínua

- Apresentação e utilização das principais ferramentas de gerenciamento de configurações e mudanças (ex.: Git, Jenkins).
- Conceito e práticas de integração contínua.
- Automatização e monitoração de processos.
- Estudo de casos aplicados ao mercado.

METODOLOGIA DE ENSINO

1. Aulas Expositivas e Dialogadas

- Apresentação dos conceitos principais da disciplina, utilizando slides, vídeos e estudos de caso reais.
- Encorajamento dos alunos a participarem ativamente, promovendo discussões em sala sobre as melhores práticas no mercado e os desafios enfrentados pelas organizações.

2. Atividades Práticas e Projetos

- Laboratórios: Realização de atividades em laboratório para que os alunos aprendam a utilizar ferramentas como Git e Jenkins, além de praticarem controle de versões e integração contínua.
- Projetos em Grupo: Desenvolvimento de um projeto prático ao longo do semestre, simulando um ambiente de trabalho em equipe, onde aplicam gestão de configurações e controle de mudanças.

3. Aprendizagem Baseada em Problemas (PBL)

- Apresentação de cenários reais ou fictícios de problemas relacionados a gestão de configurações e mudanças.
- Divisão da turma em grupos para discutir e propor soluções, utilizando conceitos e ferramentas ensinados.

4. Estudos de Caso

- Análise de estudos de caso do mercado de tecnologia para mostrar exemplos de sucesso e fracasso na gestão de configurações e mudanças.
- Análise das lições aprendidas de cada caso e como aplicar esses insights nos projetos deles.

5. Uso de Ferramentas Digitais

- Integração de plataformas digitais (ex.: GitHub, Trello) para gerenciamento de tarefas e atividades, incentivando o aprendizado prático.

RECURSOS DIDÁTICOS

- [X] Quadro
- [X] Projetor
- [X] Vídeos/DVDs
- [X] Periódicos/Livros/Revistas/Links
- [X] Equipamento de Som
- [X] Laboratório
- [X] Softwares²
- [X] Outros³

CRITÉRIOS DE AVALIAÇÃO

Realização de trabalhos práticos, projetos em equipe, exercícios de aplicação e apresentação de um projeto final integrado. A média do aluno será igual a média das notas dos projetos, variando de 0 a 100. As regras para aprovação, reprovação e final seguirão de acordo com o regimento institucional.

ATIVIDADE DE EXTENSÃO

BIBLIOGRAFIA

Bibliografia Básica:

1. PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de software: uma abordagem profissional. 9. ed. Porto Alegre: Amgh, 2021. 672 p.
2. SOMMERVILLE, Ian. Engenharia de software. Tradução de Kalinka Oliveira e Ivan Bosnic. 9. ed. São Paulo: Pearson, 2012. 529 p.
3. AQUILES, Alexandre; FERREIRA, Rodrigues. Controlando versões com Git e GitHub. São Paulo: Casa do Código, 2017. 220 p. (Série Caelum).

Bibliografia Complementar:

1. SAMPAIO, Cleiton. Qualidade de software na prática: como reduzir o custo de manutenção de software com a análise de código. Rio de Janeiro: Ciência Moderna, 2014. 204 p.
2. PRIKLADNICKI, Rafael; WILLI, Renato; MILANI, Fabiano (org.). Métodos ágeis para desenvolvimento de software. Porto Alegre: Bookman, 2014. 289 p.
3. FOWLER, Susan J. Microsserviços prontos para a produção: construindo sistemas padronizados em uma organização de engenharia de software. São Paulo: Novatec, 2017. 218 p.
4. FOWLER, Martin; BECK, Kent. Refatoração para padrões. 2. ed. Porto Alegre: Bookman, 2021. 400 p.
5. URMA, Raoul-Gabriel; WARBURTON, Richard. Desenvolvimento real de software: um guia de projetos para fundamentos em Java. Rio de Janeiro: Editora Alta Books, 2021. 174 p.

Bibliografia Suplementar:

1. CUI, Jun. Exploring the Git Flow and Docker Container Standardization on Enterprise DevOps Development Efficiency.
2. GHODKE, Gayatri Makrand; CHAVAN, Trupti. An Overview of Git. International Journal of Scientific Research in Modern Science and Technology, v. 3, n. 6, p. 17-23, 2024.
3. WOLF, Gunnar. Using the Git Version Control System to replace a Learning Management System. IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, 2024.

OBSERVAÇÕES

-
- 1 Para a oferta de disciplinas na modalidade à distância, integral ou parcial, desde que não ultrapassem os limites definidos em legislação.
2 Definido de acordo com as necessidades e especificidades da turma.
3 Definido de acordo com as necessidades e especificidades da turma.

Documento assinado eletronicamente por:

■ Alvaro Magnum Barbosa Neto, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/05/2025 15:53:26.

Este documento foi emitido pelo SUAP em 01/05/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 707922
Verificador: f32ced2b47
Código de Autenticação:



Rodovia PB 121, S/N, Centro, ESPERANÇA / PB, CEP 58135-000

<http://ifpb.edu.br> -