

Data	Conteúdo Programático	Abordagem
Aula01	<p>Apresentação Introdução</p> <ul style="list-style-type: none"> • A importância da verificação e validação de software dentro da qualidade <p>Definições e Conceitos Básicos</p> <ul style="list-style-type: none"> • O que é verificação e validação • Objetivos e benefícios das revisões em pares <p>Fundamentos Teóricos</p> <ul style="list-style-type: none"> • Verificação e Validação no CMMI 	<p>Aula expositiva e discussão em grupos Objetivos de aprendizagem</p> <ul style="list-style-type: none"> • Contextualizar as atividades de Revisão/Verificação/Validação dentro do processo de desenvolvimento de software • Compreender as áreas de processo de validação e verificação no CMMI
Aula02	<p>Planejamento de verificação e validação (V & V)</p> <ul style="list-style-type: none"> • O processo de verificação e validação • Implantando a cultura da revisão em pares e seus benefícios <p>As formas alternativas de revisões</p> <ul style="list-style-type: none"> • Inspeção • Walkthrough • Pair Programming • Revisão informal • Revisão gerencial • Revisão individual <p>Definição dos grupos de trabalhos</p>	<p>Aula expositiva e discussão em grupos Objetivos de aprendizagem</p> <ul style="list-style-type: none"> • Compreender a relação entre as fases do ciclo de desenvolvimento do software e os planos de V & V • Apresentar ao aluno as alternativas de revisão de software • Estudar as principais características e diferenças entre as formas de revisão
Aula03	<p>Walkthrough – Revisão de apresentação</p> <ul style="list-style-type: none"> • O que é uma reunião de walkthrough • Preparo da reunião • Material para a reunião • Participantes da reunião • Condução • Resultados da reunião 	<p>Aula expositiva e prática Objetivos de aprendizagem</p> <ul style="list-style-type: none"> • Analisar o processo de revisão denominado de Walkthrough e os resultados esperados • Identificar erros comuns em especificações de requisitos

Data	Conteúdo Programático	Abordagem
Aula04	Inspeção – Introdução <ul style="list-style-type: none"> • Técnica de inspeção de software • A inspeção como técnica de detecção de erros • Abordagem formal para inspeção de software • A eficácia da inspeção de software Inspeção – Planejamento <ul style="list-style-type: none"> • Papéis no processo de inspeção • Quando realizar inspeções • Agendando inspeções 	Aula expositiva e discussão em grupos <ul style="list-style-type: none"> • Apresentar ao aluno as inspeções de software • Mostrar a diferença e a complementaridade entre inspeção e teste de software • Avaliar como introduzir o processo de inspeções em uma organização
Aula05	Inspeção – Examinado o produto de trabalho <ul style="list-style-type: none"> • A Preparação • Checklists • Classificação de defeitos 	Aula expositiva e prática <ul style="list-style-type: none"> • Apresentar ao aluno como facilitar o exame de um produto • Criar um checklist como exercício em grupo • Avaliar checklists disponíveis publicamente
Aula06	Inspeção – A Reunião <ul style="list-style-type: none"> • O papel do moderador • Iniciando a reunião • Conduzindo a reunião • Avaliando o produto • Fechando a reunião 	Aula expositiva e prática Objetivos de aprendizagem <ul style="list-style-type: none"> • Entender como realizar reuniões de inspeção • Simular uma reunião de inspeção
Aula07	Inspeção – Análise de dados <ul style="list-style-type: none"> • Métricas de inspeções • Medindo o impacto das inspeções Ferramentas de Apoio ao processo de revisões <ul style="list-style-type: none"> • Ferramentas de análise estática de código • Ferramentas de gestão de incidências • Ferramentas de acompanhamento de revisões • Integração Contínua e Inspeção Contínua 	Aula expositiva e discussão em grupos Objetivos de aprendizagem <ul style="list-style-type: none"> • Apresentar características de ferramentas para apoio ao processo de revisões • Conhecer ferramentas de mercado para esse fim • Analisar os benefícios da implantação de um processo de integração e inspeção contínua do código
Aula08	Revisões em processos de desenvolvimento <ul style="list-style-type: none"> • Revisões no RUP • Revisões no MSF • Revisões e os processos ágeis e iterativos Conclusão da disciplina	Aula expositiva e discussão em grupos Entrega do artigo final. Objetivos de aprendizagem <ul style="list-style-type: none"> • Obter uma visão geral de como processos de desenvolvimento definem as atividades de revisão. • Revisar os tópicos principais vistos na disciplina

Artigos para leitura

Aula 2 - http://www.processimpact.com/reviews_book/chapter_3.pdf

Aula 3 - <http://www.yourdon.com/strucanalisys/appendices/apD.html>

Aula 4 - http://www.sei.cmu.edu/str/descriptions/inspections_body.html

Aula 5 – http://www.processimpact.com/reviews_book/peer_review_process.doc - Apenas seção “5. Inspection Procedure”

Aula 6 - <http://www2.umassd.edu/swpi/NASA/figuide.html> - Apenas seção “III. The Formal Inspection Process”

Aula 7 - <http://josepaulopapo.blogspot.com/2006/02/ferramenta-para-revises-e-inspees-de.html>
<http://www-128.ibm.com/developerworks/java/library/j-ap08016/index.html?ca=dgr-jw22SW-Inspectors>

Aula 8 - Software Quality Development and Assurance in RUP, MSF and XP - A Comparative Study

Bibliografia

1. *Peer Reviews in Software: A Practical Guide*, Karl Wieggers, Addison-Wesley, 2002.
2. *Software Inspection*, Tom Gilb and D. Graham, Addison Wesley, 1993.
3. *Engenharia de Software – Fundamentos, Métodos e Padrões*, 2ª. Edição, Wilson de Pádua Paula Filho, LTC, 2003.
4. *Engenharia de Software*, 6a. Edição, Roger S. Pressman, McGraw Hill, 2006.
5. *Code Complete*, 2nd Edition, Steve McConnell, Microsoft Press, 2004.
6. *CMMI – Guidelines for Process Integration and Product Improvement*, Mary Chrissis e outros, Addison-Wesley, 2003.
7. *Qualidade de Software*, André Koscianski e Michel dos Santos Soares, Novatec, 2006.
8. *IEEE Std. 1028 – Standard for Software Reviews and Audits*, 1998.
9. *Requirements by Collaboration*, Ellen Gottesdiener, Addison-Wesley, 2002.
10. *Applied Software Project Management*, Andrew Stellman e Jennifer Greene, O'Reilly Media, 2005.
11. *Facts and Fallacies of Software Engineering*, Robert Glass, Addison-Wesley, 2002.
12. *Quality Software Project Management*, Robert Futrell e outros, Prentice Hall, 2002.