

Nome do Componente Curricular: Laboratório Avançado de Métodos Ágeis e Software Livre

Período: Eletiva

Pré-requisitos: Algoritmos e Estruturas de Dados I - Recomendado: Programação Orientada a Objetos; Engenharia de Software

Carga Horária Total: 108h

Carga Horária Prática: 72 h

Carga Horária Teórica: 36 h

Objetivos

Gerais:

Familiarizar o estudante com metodologias ágeis de desenvolvimento de software e os mecanismos de desenvolvimento colaborativo das comunidades de software livre por meio da prática em projetos reais de software livre.

Específicos:

- Oferecer ao aluno a fundamentação sobre sistemas métodos ágeis e software livre;
- Capacitar o aluno sobre práticas de desenvolvimento de software utilizadas em ambientes ágeis;
- Capacitar o aluno a desenvolver um sistema de software real;
- Capacitar o aluno a colaborar com um projeto de software livre;
- Enfatizar aspectos avançados das metodologias, assim como no apoio e valorização dos princípios ágeis e colaboração em torno de um projeto software livre.

Ementa:

O aluno será apresentado à história do Software Livre, aos Aspectos jurídicos de software livre e aos mecanismos das comunidades de software livre. Entenderá as práticas comuns entre os Métodos Ágeis e o Software Livre. Via desenvolvimento e contribuições em projetos de software livre reais, usará as Práticas de XP (eXtreme Programming) e Desenvolvimento distribuído, apoiados por: Ferramentas para Desenvolvimento Colaborativo de Software, Boas práticas de codificação, estilo e padronização de código; Testes e integração contínua; Métricas de código-fonte.

Conteúdo Programático:

Aula 1:

Apresentação da disciplina
História do Software Livre.

Aula 2:

Aspectos jurídicos de software livre
Lista de projetos candidatos

Aula 3:

Copyright, patentes e aspectos legais
Licenças de software livre

Definição e Levantar ambientes dos projetos

Aula 4:

Finalizar ambientes dos projetos Planejamento ágil
Levantamento do backlog dos projetos

Aula 5:
Controle de versão e desenvolvimento colaborativo Início da sprint/iteração 1

Aula 6:
Continuidade da sprint/iteração 1

Aula 7:
Continuidade da sprint/iteração 1

Aula 8:
Continuidade da sprint/iteração 1

Aula 9:
Finalização da sprint/iteração 1 “Show me the code” 1

Aula 10:
Boas práticas de codificação, estilo e padronização de código Início da sprint/iteração 2

Aula 11:
Continuidade da sprint/iteração 2

Aula 12:
Continuidade da sprint/iteração 2

Aula 13:
Continuidade da sprint/iteração 2

Aula 14:
Finalização da sprint/iteração 2
Seminário “relâmpago” sobre Código Limpo

Aula 15:
Prova 1 (questões gerais sobre o projeto e práticas usadas por cada equipe)

Aula 16:
Testes e integração contínua Início da sprint/iteração 3

Aula 17:
Continuidade da sprint/iteração 3

Aula 18:
Continuidade da sprint/iteração 3

Aula 19:
Continuidade da sprint/iteração 3

Aula 20:
Finalização da sprint/iteração 3 Seminário “relâmpago” sobre Testes e CI

Aula 21:
Métricas de código-fonte Início da sprint/iteração 4

Aula 22:
Continuidade da sprint/iteração 4

Aula 23:
Continuidade da sprint/iteração 4

Aula 24:
Continuidade da sprint/iteração 4

Aula 25:
Finalização da sprint/iteração 4

Seminário “relâmpago” sobre Qualidade de código Aula 26:
Prova 2 (questões gerais sobre o projeto e práticas usadas por cada equipe)

Aula 27:
Dívida técnica e
Início da sprint/iteração 4

Aula 28:
Continuidade da sprint/iteração 4

Aula 29:
Continuidade da sprint/iteração 4
Aula 30:
Continuidade da sprint/iteração 4
Aula 31:
Finalização da sprint/iteração 4 “Show me the code” 2
Aula 32:
Apresentação final dos projetos (equipes 1 e 2)
Aula 33:
Apresentação final dos projetos (equipes 3 e 4)
Aula 34:
Apresentação final dos projetos (equipes 5 e 6)
Aula 35:
Revisão de nota Prova substitutiva
Aula 36:
Exame

Metodologia de Ensino Utilizada:

Esta unidade curricular será baseada em análise de estudos de casos e desenvolvimento de projetos reais. Os projetos serão realizados tanto em sala de aula como extra-classe, utilizando-se principalmente ferramentas de software livre. Essa unidade curricular também levará o aluno a elaborar apresentações orais e relatórios técnicos.

Recursos Instrucionais Necessários:

Quadro branco, projetor multimídia, computadores com acesso à internet, GIT, e sistema de apoio à condução da unidade curricular (Google Classroom).

CrITÉrios de Avaliação:

O sistema de avaliação será definido pelo docente responsável pela unidade curricular no início das atividades letivas devendo ser aprovado pela Comissão de Curso e divulgado aos alunos. O sistema adotado deve contemplar o processo de ensino e aprendizagem estabelecido no Projeto Pedagógico, com o objetivo de favorecer o progresso do aluno ao longo do semestre. A promoção do aluno na unidade curricular obedecerá aos critérios estabelecidos pela Pró-Reitoria de Graduação, tal como discutido no Projeto Pedagógico do Curso.

Bibliografia

Básica:

1. Kent Beck, eXtreme Programming: Explained, Addison-Wesley, 2000.
2. Martin Fowler, Refactoring: Improving the Design of Existing Code, Addison-Wesley, 2000.
3. Brian W. Kernighan and Rob Pike, The Practice of Programming, Addison-Wesley, 1998.
4. Kent Beck and Cynthia Andrés. Extreme Programming Explained: Embrace Change (2nd Edition), Addison-Wesley Professional, 2004.
5. Kent Beck and Martin Fowler. Planning Extreme Programming, Addison-Wesley Professional, 2000.

Complementar:

1. ZENKLER, Yochai. The Wealth of Networks: How Social Production Transforms Markets and Freedom. New Haven: Yale University Press. Disponível em http://cyber.law.harvard.edu/wealth_of_networks, 2006.
2. KON, Fabio; LAGO, Nelson; MEIRELLES, Paulo; SABINO, Vanessa. Software Livre e Propriedade Intelectual: Aspectos Jurídicos, Licenças e Modelos de Negócio. Capítulo no livro “XXXI Jornadas de Atualização em Informática”, SBC. 2012.
3. RAYMOND, Eric S. The Cathedral and the Bazaar. Disponível em <http://catb.org/esr/writings/cathedral-bazaar,1997-2009>.
4. STALLMAN, Richard M. Free Software Free Society: selected essays of Richard M.
5. Stallman. GNU Press. Disponível em <http://shop.fsf.org/product/free-software-free-society>, 2002.