

MC437 – Projeto de Sistemas de Informação

2º Semestre/2016

Plano Geral da Disciplina

Profa.: Eliane Martins

Horários da disciplina:

- | | |
|----------------------------|------------------|
| ■ Teórica: 6ª. das 19h-21h | Local: CC52 |
| ■ Lab: 6as. das 21-23h | Local: CC02/CC03 |

Onde me achar:

eliane_at_ic.unicamp.br
IC1 - sala 36

Monitor: Jael Zela Ruiz (PED)

jzelar@gmail.com

Horário de atendimento:

Enviar email para marcar horário

Apresentação do curso

EMENTA

- Análise e Projeto de Sistemas de Informação usando técnicas de Bancos de Dados, Interface Humano-Computador e Engenharia de Software

PROGRAMA

- Estudos: desenvolvimento na Web
- Levantamento de requisitos funcionais e não funcionais
- Projeto arquitetural
- Desenvolvimento e testes dos componentes do sistema.
- Integração e testes dos componentes do sistema

Atividades do Curso

Para a realização do programa será utilizado um estudo de caso real, que será desenvolvido usando métodos ágeis. O objetivo é fazer com que o aluno aplique as boas práticas da Engenharia de Software, sem perda de agilidade no processo de desenvolvimento.

Devido a características intrínsecas como capacidade de interoperar com outros sistemas e o uso de padrões abertos (ex.: HTTP, XML), os serviços Web têm tido grande interesse tanto na academia quanto na indústria. Por essa razão, o enfoque da disciplina este ano será no desenvolvimento de aplicações orientadas a serviços. Mais especificamente, vamos focar as questões de segurança de aplicações orientadas a serviço, e como especificar e implementar mecanismos de segurança utilizando métodos ágeis.

- Os trabalhos serão feitos em grupos. O número de membros de cada grupo será definido de acordo com o tamanho da turma. Todo aluno deve pertencer a um grupo: **NÃO SERÃO ACEITOS TRABALHOS INDIVIDUAIS.**

SÓ SERÁ PERMITIDO DESENVOLVER CÓDIGO DURANTE A AULA. A equipe não pode trazer código pronto, nem inseri-lo no repositório em horário fora da aula

A seguir, as atividades a serem realizadas durante a disciplina.

Apresentações

Durante a disciplina serão feitas apresentações sobre temas ligados ao projeto. As apresentações serão feitas pelo grupo. Maiores informações sobre conteúdo e datas das apresentações serão dadas no decorrer da disciplina.

Entregas

O desenvolvimento é incremental, dividido em sprints. Serão dois sprints no total. A cada sprint, diversas iterações são realizadas para desenvolver o incremento. A cada iteração, diversos artefatos serão produzidos, e deverão ser entregues.

Todas as entregas deverão ser depositados no repositório do projeto ao final de cada aula. O repositório utilizado será indicado no decorrer da disciplina.

Questionários

Serão passados dois questionários: um no início da disciplina, para coleta de informações que nos permita realizar a divisão em grupos; o outro no final, para obter a avaliação individual pelos pares.

Critério de Avaliação

Cálculo da média:

A média **M** do aluno será calculada pela soma das notas abaixo:

- Apresentações: peso 1
- Sprint1: peso 3
- Sprint2: peso 3
- PAC (empacotamento do produto): peso 1
- Questionários: peso 1
- Avaliação pelos pares: peso 1

Dentro de cada sprint, as notas são divididas em **nota individual** e **nota do grupo**. As indicações dos entregáveis individuais e de grupo serão definidos no decorrer da disciplina.

Observação:

- **Não** há provas e nem exame nessa disciplina.

Frequência às aulas

- O limite de faltas é de **25%** do total das aulas previstas. O aluno que tiver menos do que **75%** de presença é **reprovado por falta**.
- **Única Exceção:** O aluno com problemas de saúde que impeçam o seu comparecimento às aulas por períodos mais longos, devem entrar com um requerimento de "regime especial" junto à Diretoria Acadêmica (DAC).
- A presença deve ser documentada com uma assinatura em uma lista de chamada para posterior contabilização. **A assinatura nas listas de chamada deve ser a mesma utilizada em documentos oficiais** junto à Diretoria Acadêmica (DAC). **Não são aceitáveis iniciais apenas ou rubricas.**

Conduta Ética

- Os trabalhos deverão ser de autoria do aluno ou do grupo **UNICAMENTE**. Discussões e troca de idéias com colegas, professor ou assistente são saudáveis e bem vindas, mas a solução final deve ser exclusivamente do autor, ou dos autores, quando for trabalho em grupo.
- Consultas a fontes externas (Web, artigos e livros) são válidas e altamente recomendadas desde que explicitamente referenciados no trabalho.
- Qualquer outro tipo de conduta será considerado como plágio, e implicará em pontuação **ZERO** para todos os envolvidos.

Referências

- Orit Hazzan and Yael Dubinsky. Agile Software Engineering (Undergraduate Topics in Computer Science). Springer London; 1 edition (February 28, 2009)
- Ken Schwaber and Mike Beedle. Agile Software Development with Scrum. Prentice Hall, Upper Saddle River, New Jersey, 2002.
- Mary Poppendieck and Tom Poppendieck. Lean Software Development: An Agile Toolkit. Addison-Wesley, 2003.
- Mike Cohn. User Stories Applied: For Agile Software Development. Addison-Wesley, 2004.

- John F. Smart. BDD In Action. Manning, 2013.

Recursos na Internet

(Esta lista poderá ser atualizada ao longo do curso. Ela está longe de ser exaustiva)

- Sobre UML: www.uml.org
- Sobre arquitetura de software: <http://www.serc.nl/people/florijn/interests/arch.html>
- Padrões e Anti-padrões:
http://www.dmoz.org/Computers/Programming/Methodologies/Patterns_and_Anti-Patterns/
- Agile Manifesto – <http://agilemanifesto.org/iso/ptbr/>
- Scrum Guide - versão de 2011 em português - [http://www.scrum.org/storage/Scrum Guide 2011 - PTBR.pdf](http://www.scrum.org/storage/Scrum%20Guide%202011-PTBR.pdf)
- Scrum Master in Under 10 Minutes video
<http://www.axosoft.com/ontime/videos/scrum/#scrum-diagram>
- Scott Ambler. Introduction to TDD: <http://www.agiledata.org/essays/tdd.html>
- Ryan Greenhall. BDD by Examples: <http://www.ryangreenhall.com/articles/bdd-by-example.html>
- John Smart tem vários tutoriais:
http://pt.slideshare.net/search/slideshow?searchfrom=header&q=john+smart%2C+bdd&ud=any&ft=all&lang=**&sort= Em especial:
 - sobre TDD:
http://pt.slideshare.net/skillsmatter/real-developersdontneedunittests?qid=5a746836-4918-43ef-b7cc-09881e2160d4&v=default&b=&from_search=12
 - BDD para testes Web:
http://pt.slideshare.net/wakaleo/bdd-atdd-page-objects-the-road-to-sustainable-web-testing?qid=5a746836-4918-43ef-b7cc-09881e2160d4&v=default&b=&from_search=8
- Sobre Cucumber com JVM:
 - <https://cucumber.io/docs/reference/jvm>
- Sobre Selenium:
 - <http://www.seleniumhq.org/projects/webdriver/>
- Sobre serviços Web e aspectos de segurança:
<http://gcseg.das.ufsc.br/wssec/pubs/mello06-sbseg-mc-seg-ws.pdf>
- Guia: OWASP Secure Coding Practices
 - https://www.owasp.org/images/b/b3/OWASP_SCP_v1.3_pt-BR.pdf
- Sobre Behavior Driven Security Testing:
<https://netsense.ch/download/pentestmag-may2012-behaviour-driven-security-testing.pdf>
- BDD-Security: permite descrever cenários para especificar/testar segurança
<https://www.continuumsecurity.net/bdd-intro.html>