

PLANO DE CURSO

UNIDADE RESPONSÁVEL
DEPARTAMENTO DE
ENGENHARIA DE COMPUTAÇÃO
E AUTOMAÇÃO
CÓDIGO
DCA0205
DOCENTES PROPONENTES

Ronildo Pinheiro de Araujo Moura

NOME DO COMPONENTE
PROJETO E ENGENHARIA DE SOFTWARE

CARGA HORÁRIA
45h

1. Insira as informações aqui.
2. Insira as informações aqui.
3. Insira as informações aqui.

CONTEÚDO

Ciclo de vida de software. Metodologias para desenvolvimento de software: abordagens clássicas e abordagens ágeis. Análise de requisitos. Arquiteturas de software: monolítico e micro-serviços. Projeto de software. Técnicas de teste de software (testes unitários, de regressão, operacionais, funcionais, de desempenho). Documentação e experiência do usuário.

METODOLOGIA

O material didático, expositivo, para as aulas foram preparados para um curso remoto assíncrono, e pode incluir notas de aula, slides, slides-comentados, sugestão de leituras e/ou vídeos-aulas.

PROCEDIMENTOS DE AVALIAÇÃO DA APRENDIZAGEM

O processo de avaliação será realizado acompanhando o processo de aprendizagem durante todo o semestre. Dito isso, o processo de avaliação utilizará três tipos diferentes de atividades. Uma atividade individual de retenção de aprendizagem será apresentada ao término de cada aula. Alguns problemas de produção de software serão discutidos durante o curso. Desta forma, algumas atividades individuais de produção de software serão propostas. Por fim, um projeto final em grupo deve ser entregue.

A submissão da atividade de retenção são via SIGAA. A submissão das atividades de produção de software são via Github Classroom. A submissão do do trabalho final será feita via SIGAA e o acompanhamento das etapas via Github Classroom.

DETALHAMENTO DOS RECURSOS DIDÁTICOS

Os recursos didáticos utilizados durante este curso foram divididos em três aspectos de ensino: material de aula (tanto expositivo quanto ao encontro virtual), comunicação com e entre os alunos, e por fim, das atividades de avaliação.

O material das aulas foram preparados para um curso remoto assíncrono, composto por notas de aula, slides e pequenos vídeos cobrindo os tópicos da aula. Todos os arquivos serão postados no SIGAA, na página da disciplina. Dessa forma, os alunos devem possuir acesso a um computador com internet. Em cada uma das aulas será disponibilizado um roteiro que relaciona e organiza cada um dos arquivos e atividades disponíveis. Além disso, dois encontros virtuais ocorrerão a cada semana para que os alunos possam tirar dúvidas e para que algumas discussões sobre o conteúdo possam ser realizadas. Os encontros virtuais serão realizados na ferramenta Google Meet(<https://meet.google.com/>).

A principal ferramenta de comunicação será o SIGAA, bem como o fórum da disciplina, no SIGAA, para discussões relacionadas ao curso. Todas as questões sobre os materiais e conteúdos do curso devem ser postados no fórum da disciplina. Desta forma, todos os alunos são beneficiados por essas discussões.

As atividades de avaliação utilizaram o SIGAA para atividade de retenção, enquanto os problemas de produção de software serão acompanhadas na plataforma Github Classroom(<https://classroom.github.com>).

O trabalho final usará o SIGAA e o Github Classroom.

Observa-se que todas as ferramentas extras ao SIGAA são gratuitas e contam tanto com ambiente web quanto com aplicativos para computadores e celulares.

CRITÉRIOS PARA A REALIZAÇÃO DAS ATIVIDADES E VALIDAÇÃO DA ASSIDUIDADE DOS DISCENTES

Como o curso é assíncrono, a presença aos encontros virtuais é facultativa. Os vídeos desses encontros também serão postados no SIGAA.

A validação da assiduidade ocorrerá através dos relatórios de acesso aos conteúdo de ensino disponível pelo SIGAA.

CRONOGRAMA

DATA	CONTEÚDO	RECURSOS DIDÁTICOS
15/06	Apresentação da Disciplina e Instruções sobre o Projeto Final	Video Aula
16/06	Introdução a ES e Conceitos Fundamentais	Video Aula + Slide
17/06	Processo de Desenvolvimento de Software	Video Aula + Slide ou Texto Suplementar
18/06	Modelos de Processo de Software	Encontro Virtual + Texto Suplementar
19/06		

22/06	Abordagens Ágeis	Video Aula + Slide ou Texto Suplementar
23/06	Engenharia de Requisitos: Documento de Requisitos, Requisitos de Usuário, Requisitos de Sistema	Video Aula + Slide ou Texto Suplementar
24/06	Introdução UML	Video Aula + Slide ou Texto Suplementar
25/06	UML: Diagramas Estruturais, Diagramas Comportamentais	Encontro Virtual + Texto Suplementar
26/06		
29/06	Engenharia de Requisitos: Documento de casos de Uso, Casos com cenários	Video Aula + Slide ou Texto Suplementar
30/06	Arquitetura de Software, Padrões de Arquitetura	Video Aula + Slide ou Texto Suplementar
01/07	Arquitetura de Software, Tipos de Arquitetura	Video Aula + Slide ou Texto Suplementar
02/07	Projeto de Arquitetura	Encontro Virtual + Texto Suplementar
03/07		
06/07	Apresentação da Primeira Parte do Projeto	Encontro Virtual
07/07	Projeto de Software Orientado a Objetos, UML: Diagrama de Classes	Video Aula + Slide ou Texto Suplementar
08/07	UML: Diagrama de Sequência, Diagrama de Comunicação, Diagrama de Atividades	Video Aula + Slide ou Texto Suplementar
09/07	Projeto de Software com Reúso.	Encontro Virtual + Texto Suplementar
10/07		
13/07	Padrões de Projeto	Video Aula + Slide ou Texto Suplementar
14/07	Introdução a Teste de Software	Video Aula + Slide ou Texto Suplementar
15/07	Teste de Software: Teste de Unidade, Teste de Regressão	Video Aula + Slide ou Texto Suplementar
16/07	Teste de Software: Testes Operacionais, Testes Funcionais., Testes de desempenho.	Encontro Virtual + Texto Suplementar
17/07		
20/07	Introdução a Qualidade de Software	Video Aula + Slide ou Texto Suplementar
21/07	Experiência do Usuário	Video Aula + Slide
22/07	(Aula extra) DevOps	Video Aula + Slide
23/07		
24/07	Entrega do Trabalho Final	Encontro Virtual

HORÁRIOS DE ATENDIMENTO

2ª 14:00~16:00	3ª 14:00~16:00	4ª 14:00~16:00	5ª 14:00~16:00	6ª
-------------------	-------------------	-------------------	-------------------	----

REFERÊNCIAS

SOMMERVILLE, Ian. **Engenharia de Software**. 9.ed. São Paulo: Pearson, 2011. 529 p. ISBN: 9788579361081.

LARMAN, Craig. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3.ed. Porto Alegre: Bookman, 2007. xiv, 695 p. ISBN: 9788560031528.

MALDONADO, José Carlos; JINO, Mario; DELAMARO, Marcio Eduardo (Org). **Introdução ao teste de software**. Rio de Janeiro: Elsevier Campus, 2007. 394 p. (Série Campus SBC) ISBN: 9788535226348.

GAMMA, Erich et al. **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000. 364 p. ISBN: 8573076100.

MARTIN, Robert C. **The clean coder: a code of conduct for professional programmers**. Upper Saddle River, NJ: Prentice Hall, c2011. xxii, 210 p. ISBN: 0137081073.