



Universidade Federal do Ceará
Campus de Russas

PLANO DE ENSINO DE DISCIPLINA

Ano/Semestre
2022/2

1. Identificação				
1.1. Unidade: Campus Russas				
1.2. Curso: Ciência da Computação				
1.3. Estrutura Curricular (ano-período): 2018.2				
1.4. Nome da Disciplina: Engenharia de Software				
1.5. Código da Disciplina: RUS0084				
1.6. Caráter da Disciplina: (X) Obrigatória () Optativa				
1.7. Regime de Oferta da Disciplina: (X) Semestral () Anual () Modular				
1.8. Carga Horária (CH) Total:	C.H. Teórica:	C.H. Prática:	C.H. EaD:	C.H. Extensão:
64H	64H	-	-	-
1.9. Pré-requisitos (quando houver): Programação Orientada a Objetos				
1.10. Co-requisitos (quando houver):				
1.11. Equivalências (quando houver):				
1.12. Professor(es): Anna Beatriz dos Santos Marques				
2. Justificativa				
Introduzir através do estudo dos conceitos de engenharia de software os conhecimentos básicos necessários para entender o funcionamento dos processos de software, elicitação de requisitos e gerência de projetos.				
3. Ementa				
Visão geral e introdutória dos princípios fundamentais e éticos-profissionais da Engenharia de Software. Introdução às atividades de engenharia de requisitos; projeto de software; modelos de desenvolvimento; e conceitos de gerenciamento de projetos (qualidade, estimativa de custo, etc.) na engenharia de software				
4. Objetivos – Geral e Específicos				

ATENÇÃO! As informações a serem preenchidas neste formulário devem ser exatamente iguais às constantes no formulário de criação/regulamentação da disciplina aprovado pela Câmara de Graduação.

Objetivo Geral:
(i) o aluno deve compreender os conceitos fundamentais de Engenharia de Software.
Objetivos Específicos:
(i) fornecer uma visão geral de Engenharia de Software
(ii) levantar requisitos e validá-los;
(iii) elaborar um documento de requisitos;
(iv) explicar, analisar e modelar processos de desenvolvimento de software;
(v) fornecer conceitos básicos de gerência de projetos; e
(vi) listar possíveis documentos utilizados em um processo de software.

5. Calendário de Atividades		
Data	Descrição do Conteúdo	Carga Horária
16/08/22	Introdução à engenharia de software <i>Apresentação da ementa e cronograma de provas e trabalhos</i>	2h
18/08/22	Princípios fundamentais e éticos-profissionais da Engenharia de Software	2h
23/08/22	Soft skills na Engenharia de Software <i>Definição de equipes para os seminários e sorteio de temas (artigos)</i>	2h
25/08/22	Processos de Software	2h
30/08/22	Modelos de Processos Tradicionais e RUP	2h
01/09/22	Modelos de processos ágeis - princípios	2h
06/09/22	Aula cancelada	2h
08/09/22	Sem aula <i>Participação em Oficina da CGI.Br</i>	-
13/09/22	Aula cancelada	2h
15/09/22	Modelos de processos ágeis – XP	2h
20/09/22	Modelos de processos ágeis – Scrum	2h
22/09/22	Seminário – apresentação de artigos sobre Processos e Fatores Humanos	2h
27/09/22	Requisitos – Conceitos e Tipos	2h
29/09/22	Técnicas de Levantamento de Requisitos – Personas	2h
04/10/22	Engenharia de Requisitos <i>SBES 2022</i>	2h
06/10/22	Especificação de Requisitos <i>SBES 2022</i>	2h
11/10/22	Técnicas de Levantamento de Requisitos – Brainstorming com USARP <i>Explicação sobre o Trabalho Prático 1</i>	2h
13/10/22	Seminário – apresentação de artigos sobre Requisitos	2h
18/10/22	Execução do Trabalho Prático 1	2h

ATENÇÃO! As informações a serem preenchidas neste formulário devem ser exatamente iguais às aquelas constantes no formulário de criação/regulamentação da disciplina aprovado pela Câmara de Graduação.

	<i>IHC 2022</i>	
20/10/22	Execução do Trabalho Prático 1 <i>IHC 2022</i>	2h
25/10/22	Encerramento da gamificação e apresentação dos resultados	2h
27/10/22	Entrega do Trabalho Prático 1	2h
01/11/22	Prova Parcial 1	2h
03/11/22	Projeto de Software	2h
08/11/22	Projeto Orientado a Objetos com UML <i>SBQS 2022</i>	2h
10/11/22	Projeto Orientado a Objetos com UML <i>SBQS 2022</i>	2h
15/11/22	Feriado Nacional – Proclamação da República	-
17/11/22	Projeto de interface <i>CBIE 2022</i>	2h
22/11/22	Revisão e exercícios sobre projeto de software	2h
24/11/22	Encontros Universitários	2h
29/11/22	Seminário – apresentação de artigos sobre Design e Processos	2h
01/12/22	Prova Parcial 2	2h
06/12/22	Entrega do Trabalho Prático 2	2h
08/12/22	Prova de 2ª chamada	2h
13/12/22	Aula extra – se necessário	-
15/12/22	Avaliação Final	-
22/12/22	Consolidação das sínteses de notas e frequências 2022.2 das disciplinas semestrais e anuais pelos professores – Prazo Final	-

6. Metodologia de Ensino
Aula expositiva e aula dialogada para abordar conteúdos introdutórios e teóricos sobre Engenharia de Software; Aprendizagem baseada em projetos para a realização de atividades práticas do processo de desenvolvimento de software; Seminário para apresentação sobre Requisitos, Design e Gestão de Projetos.
7. Atividades Discentes
Assiduidade às aulas. Participação do aluno no desenvolvimento das aulas. Preparação de seminários sobre os conteúdos abordados. Execução de trabalhos para elaboração de documento de requisitos e artefatos de projeto de software. Provas subjetivas e objetivas.
8. Sistema de Avaliação
Conforme o Regimento Geral da UFC, a avaliação de rendimento do aluno far-se-á segundo os critérios de assiduidade e eficiência. Na verificação da assiduidade será aprovado o aluno que frequentar 75% (setenta e cinco por cento) ou mais da carga horária da disciplina, vedado o abono de faltas. A verificação da eficiência compreenderá, no mínimo, duas avaliações progressivas e uma

ATENÇÃO! As informações a serem preenchidas neste formulário devem ser exatamente iguais àquelas constantes no formulário de criação/regulamentação da disciplina aprovado pela Câmara de Graduação.

avaliação final. Será aprovado por média o aluno que apresentar média aritmética das notas resultantes das avaliações progressivas igual ou superior a 07 (sete). O aluno que apresentar a média igual ou superior a 04 (quatro) e inferior a 07 (sete), será submetido à avaliação final. Nesse caso, o aluno será aprovado quando obtiver nota igual ou superior a 04 (quatro) na avaliação final e média final igual ou superior a 05 (cinco).

Serão aplicados dois critérios de avaliação, são eles: a frequência e notas. O aluno com frequência maior ou igual a 75% (setenta e cinco por cento) do total de aulas ministradas no curso atende ao primeiro critério.

As notas da disciplina são compostas de duas Avaliações Progressivas (AP1 e AP2).

A Avaliação Progressiva 1 (AP1) é composta de:

- Prova Parcial 1 (PP1): Prova escrita (0,0 – 5,0)
- Exercícios 1 (EXE1): Exercícios realizados durante as aulas e relacionados ao conteúdo da Prova Parcial 1 (0,0 – 1,0)
- Trabalho Prático 1 (TP1): Aplicação de técnicas de engenharia de requisitos (0,0 – 4,0)

A Avaliação Progressiva 2 (AP2) é composta de:

- Prova Parcial 2 (PP2): Prova escrita (0,0 – 4,0)
- Trabalho Prático 2 (TP2): Elaboração de artefatos de projeto de software (0,0 – 3,0)
- Seminário (SEM): Apresentação de artigos científicos sobre os conteúdos abordados (0,0 – 3,0)

Art. 114 (Regimento UFC). Na verificação da eficiência, será aprovado por média o aluno que, em cada disciplina, apresentar média aritmética das notas resultantes das avaliações progressivas igual ou superior a 07 (sete).

§ 1º O aluno que apresentar a média de que trata o caput deste artigo, igual ou superior a 04 (quatro) e inferior a 07 (sete), será submetido à avaliação final.

A partir de sua Média APs, o aluno deverá observar as seguintes condições:

Se Média das APs < 4.0 = o aluno estará reprovado.

Se Média das APs >= 7.0 = o aluno estará aprovado.

Se Média das APs >= 4.0 e Média das APs < 7.0, o aluno deverá fazer a Avaliação Final (AF) e sua Nota Final será calculada da seguinte forma:

$$Nota\ Final = \frac{Media\ APs + AF}{2}$$

Se o aluno obtiver a nota da Avaliação Final <4.0, ele estará reprovado;

Se o aluno obtiver Nota Final <5.0, ele estará reprovado, caso contrário, estará aprovado.

9. Bibliografia Básica e Complementar

Bibliografia Básica:

1. SOMMERVILLE, I. Engenharia de software. 9 ed. Addison Wesley, 2011. ISBN: 9788579361081.
2. PRESSMAN, R. Engenharia de software. 6 ed. Pearson, 2009.
3. LARMAN, Craig. Utilizando UML e padroes : uma introducao a analise e ao projeto orientados a objetos. 5. ed. Porto Alegre: Bookman, 2007. 695 p. ISBN 856003152-9.

ATENÇÃO! As informações a serem preenchidas neste formulário devem ser exatamente iguais àquelas constantes no formulário de criação/regulamentação da disciplina aprovado pela Câmara de Graduação.

Bibliografia Complementar:

1. TELES, V. Extreme programming. Novatec. 2004. ISBN: 8575220470
2. MOLINARI, L. Gerência de configuração: técnicas e práticas no desenvolvimento do software. VISUAL BOOKS. ISBN: 8575022105
3. DELAMARO, M.; MALDONADO, J.C.; JINO, M. Introdução ao teste de software. Campus, 2007.
4. KERIEVSKY, J. Refatoração para padrões. Bookman, 2008. ISBN:9788577802449
5. PEZZÉ, M.; YOUNG, M. Teste e análise de software: processos, princípios e técnicas. Bookman, 2008. ISBN: 9788577802623
6. PILONE, D.; MILES, R. Use a cabeça! desenvolvimento de software. ALTA BOOKS, 2008.

10. Parecer

Assinatura do Professor

____/____/____ _____

Aprovação da Coordenação do Curso

____/____/____ _____
Coordenador do Curso

Aprovação da Coordenação Acadêmica

____/____/____ _____
Coordenadora Acadêmica

ATENÇÃO! As informações a serem preenchidas neste formulário devem ser exatamente iguais àquelas constantes no formulário de criação/regulamentação da disciplina aprovado pela Câmara de Graduação.