



UNIVERSIDADE FEDERAL DO CEARÁ
PRÓ-REITORIA DE GRADUAÇÃO
COORDENADORIA DE PROJETOS E ACOMPANHAMENTO CURRICULAR
DIVISÃO DE DESENVOLVIMENTO CURRICULAR

FORMULÁRIO PARA CRIAÇÃO E/OU REGULAMENTAÇÃO DE DISCIPLINA

() **Regulamentação** (se a disciplina está prevista no Projeto Pedagógico)

() **Criação/Regulamentação** (se a disciplina não está prevista no Projeto Pedagógico)

1. Unidade Acadêmica que oferta a Disciplina (Faculdade, Centro, Instituto, *Campus*):
Campus Quixadá

2. Departamento que oferta a Disciplina (quando for o caso):

--

3. Curso(s) de Graduação que oferta(m) a disciplina

Código do Curso	Nome do Curso	Grau do Curso ¹	Currículo (Ano/Semestre)	Caráter da Disciplina ²	Semestr e de Oferta ³	Habilitação ⁴
402	Engenharia de Software	Bacharelado	2010.1	Optativa	7º	--
404	Ciência da Computação	Bacharelado	2013.1	Optativa	7º	--

4. Nome da Disciplina:

Reuso de Software

5. Código da Disciplina (preenchido pela PROGRAD):

QXD0068

6. Pré-Requisitos	Não ()	Sim (X)	
		Código	Nome da Disciplina/Atividade
		QXD0058 (402)	Projeto Detalhado de Software
		QXD0019 (404)	Engenharia de Software

7. Correquisitos	Não (X)	Sim ()	
		Código	Nome da Disciplina/Atividade

¹ Preencher com *Bacharelado, Licenciatura* ou *Tecnólogo*.

² Preencher com *Obrigatória, Optativa* ou *Eletiva*.

³ Preencher quando obrigatória.

⁴ Quando eletiva, preencher com a habilitação ou ênfase a que se vincula a disciplina.

8. Equivalências	Não (X)	Sim ()	
		Código	Nome da Disciplina/Atividade

9. Turno da Disciplina (é possível marcar mais de um item):
 Matutino Vespertino Noturno

10. Regime da Disciplina:
 Semestral Anual Modular

11. Justificativa para a criação/regulamentação desta disciplina – Máximo de 500 caracteres
(mostrar a importância da área / do conteúdo para a formação do aluno, a pertinência da disciplina na integralização curricular e outros aspectos):

O crescente avanço tecnológico tornou possível a construção de sistemas computacionais cada vez maiores, complexos e onipresentes que colaboram para prover serviços essenciais à sociedade. Nesse tipo de sistema, o software assume um papel fundamental devendo ser capaz de operar em ambientes distribuídos, embarcados e heterogêneos; adaptar-se às mudanças do ambiente de execução; e comportar-se de maneira fidedigna. Alinhar o gerenciamento da complexidade às restrições de tempo, custo e qualidade demandadas pelas partes interessadas dentro de um projeto de software, representa um dos principais desafios ao desenvolvimento de software. O reuso de software é uma subárea da Engenharia de Software que pode ser entendida como o uso de software existente, ou conhecimento (de projeto ou codificação) referente a esse software, para construir um novo software. Dessa forma, com a reutilização de artefatos de software previamente desenvolvidos, testados e certificados, os custos no desenvolvimento de novos sistemas de software tendem a ser reduzidos e os indicadores de qualidade e produtividade, aumentados. Além disso, a adoção de abordagens sistemáticas de reuso de software, como Linhas de Produto de Software, possibilitam que as organizações sejam guiadas tanto na construção de novos sistemas de software, a partir de artefatos reutilizáveis (desenvolvimento com reutilização), quanto na construção destes artefatos em si (desenvolvimento para reutilização). Dentro da matriz curricular do curso, a disciplina de Reuso de Software, além de apresentar novas técnicas e abordagens de reuso de software sob uma perspectiva integrada e sistêmica, busca aprofundar conceitos básicos de reutilização introduzidos nas disciplinas de “Programação Orientada a Objetos”, “Projeto Detalhado de Software” e “Arquitetura de Software”. Além do aprofundamento conceitual, a disciplina prioriza atividades teórico/prático associadas à construção de artefatos reutilizáveis (ex. bibliotecas, componentes e frameworks), leitura e compreensão do formato padrão utilizado por engenheiros de software para documentar boas decisões de projeto para uma futura reutilização desse conhecimento (ex., padrões de projeto e frameworks de arquitetura) e como sistematizar o emprego do conhecimento de projeto e de artefatos de software pré-existentes na construção de novos sistemas de software (ex., frameworks de aplicação, programação baseada em componentes, programação orientada a aspectos e linha de produtos de software).

12. Objetivo(s) da Disciplina:

Objetivos Gerais:

- Compreender as principais técnicas e abordagens de reuso de software;
- Planejar, implantar, executar e avaliar o reuso de software dentro de uma organização;
- Identificar oportunidades para a utilização de reuso de software.

Objetivos específicos:

- Documentar e utilizar padrões de projeto;
- Utilizar componentes, bibliotecas e frameworks de aplicação;
- Gerenciar artefatos de software reutilizáveis;
- Modelar processos de reutilização de software;
- Desenvolver artefatos de software de forma apropriada para que possam ser reutilizados;
- Avaliar a qualidade dos artefatos reutilizáveis;
- Mensurar o custo do reuso de software para a organização.

13. Ementa:

Conceitos básicos e importância de Reuso de Software. Principais abordagens de reuso: Engenharia de Domínio, Linha de Produtos de Software, Desenvolvimento Baseado em Componentes, Arquitetura Orientada a Serviços, Padrões e Frameworks.

14. Descrição da Carga Horária

Número de Semanas:	Número de Créditos:	Carga Horária Total:	Carga Horária Teórica:	Carga Horária Prática:
16	4	64	32	32

15. Bibliografia Básica (sugere-se a inclusão de, pelo menos, 03 títulos):

MILI, Hafedh. Reuse-based software engineering: techniques, organization and measurement. New York: Wiley, 2002. 636 p. ISBN 0471398195.

EZRAN, M.; MORISIO, M.; TULLY, C. Practical software reuse. Berlim: Springer, 2002

POHL, K.; BÖCKLE, G.; LINDEN, F. J. Software product line engineering: foundations, principles and techniques. Berlim: Springer, c2010. xxvi, 467 p. ISBN 9783642063640 (broch.)..

16. Bibliografia Complementar (sugere-se a inclusão de, pelo menos, 05 títulos – de acordo com instrumento de avaliação de Curso de Graduação, INEP/maio-2012 ou legislação posterior):

SOMMERVILLE, I. Engenharia de software. 7. ed. São Paulo: Pearson Addison-Wesley, 2007.

PRESSMAN, R. Engenharia de software. 6. ed. São Paulo: Mc Graw-Hill, 2006. ISBN 8586804576

GAMMA, E.; HELM, JOHNSON, R.; R.; VLISSIDES, J. Padrões de projeto: soluções reutilizáveis de software orientado a objetos. Porto Alegre: Bookman, 2000. 364 p.

BRAUDE, Eric J.; BERNSTEIN, Michael E. Software engineering: modern approaches. 2nd ed. Hoboken, New Jersey: J. Wiley & Sons, 2011. xvi, 782 p.

BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. Software architecture in practice. 3rd ed. Upper Saddle River, NJ: Addison-Wesley, c2013. xix, 589 p. (SEI series in software engineering). ISBN 9780321815736 (enc.).

ALMEIDA, Eduardo Santana de; ALVARO, Alexandre; GARCIA, Vinicius Cardoso; MASCENA, Jorge Cláudio Cordeiro Pires; BURÉGIO, Vanilson André de Arruda; NASCIMENTO, Leandro Marques do; LUCREDIO, Daniel; e MEIRA, Silvio Lemos. C.R.U.I.S.E - Component Reuse In Software Engineering. Recife: Cesar ebooks, 2007. Disponível em:

<<http://www.repositoriobib.ufc.br/000023/000023bd.pdf>>. Acesso em: 28 jan. 2016. [recurso eletrônico]

...

KANG, K. C.; SUGUMARAN, V.; PARK, S. Applied software product line engineering. Boca Raton, Florida: CRC Press, 2010.

17. Aprovação do Colegiado do Departamento (quando for o caso)

Data de Aprovação:

Chefe(a) do Departamento
Assinatura e Carimbo

18. Aprovação do(s) Colegiado(s) de Curso(s)

Código do Curso:	Data de Aprovação:	<hr/> Coordenador(a) do Curso Assinatura e Carimbo
Código do Curso:	Data de Aprovação:	<hr/> Coordenador(a) do Curso Assinatura e Carimbo

19. Aprovação do Conselho da Unidade Acadêmica

Data de Aprovação:	<hr/> Diretor(a) da Unidade Acadêmica Assinatura e Carimbo
---------------------------	---

20. Aprovação do Conselho de Ensino, Pesquisa e Extensão (Câmara de Graduação)

Data de Aprovação:	<hr/> Presidente(a) da Câmara de Graduação Assinatura e Carimbo
---------------------------	--

Orientações para tramitação do processo:

Deve ser aberto e encaminhado processo à Pró-Reitoria de Graduação / Câmara de Graduação, contendo: 1) Ofício(s) informando a data de aprovação da criação e/ou regulamentação da(s) disciplina(s) pela Coordenação do Curso, pelo(s) Departamento(s) envolvido(s) – se for o caso – e pela Direção da Unidade Acadêmica; 2) Formulário para Criação e/ou Regulamentação de Disciplina integralmente preenchido, com assinaturas, datas e carimbos solicitados.

ANEXO - Descrição do Conteúdo e Carga Horária

12. Descrição do Conteúdo e Carga Horária					
Unidades e Assuntos das Aulas			Nº de Horas Teóricas	Nº de Horas Práticas	Nº de Horas EaD (quando for o caso):
1. Introdução ao Reuso de Software 1.1. Reuso de Software e a Engenharia de Software 1.2. Estado da Arte e da Prática em Reuso de Software 1.3. Aspectos Gerais do Reuso de Software			4h	4h	
2. Aspectos Organizacionais 2.1. Organização do Reuso de Software 2.2. Serviços de Suporte ao Reuso 2.3. Institucionalização do Reuso 2.4. Barreiras ao Reuso			4h	2h	
3. Engenharia de Domínio 3.1. Construindo Artefatos Reutilizáveis 3.2. Análise de Domínio 3.3. Paradigmas de Programação e Reutilização			6h	6h	
4. Engenharia de Domínio Orientada a Objetos 4.1. Abstração e Parametrização Orientada a Objetos 4.2. Técnicas de Composição e Orientação a Objetos 4.3. Padrões de Projeto 4.4. Frameworks de Aplicação			4h	6h	
5. Engenharia de Aplicação 5.1. Paradigmas e Ciclo de Vida 5.2. Armazenamento e Recuperação de Componentes 5.3. Integração de Artefatos Reutilizáveis			6h	6h	
6. Aspectos Gerenciais do Reuso de Software 6.1. Métricas de Reuso de Software 6.2. Estimativa de Custos de Reuso de Software			4h	4h	
7. Tecnologias de Reuso de Software 7.1. Engenharia de Software Baseada em Componentes 7.2. Engenharia de Software Baseada em Serviços 7.3. Linha de Produtos de Software			4h	4h	
Número de Semanas:	Número de Créditos:	Carga Horária Total:	Carga Horária Teórica:	Carga Horária Prática:	Carga Horária EaD:
16	4	64	32	32	--