



Universidade Federal do Ceará
Campus de Russas

PLANO DE ENSINO DE DISCIPLINA

Ano/Semestre

2023/01

1. Identificação				
1.1. Unidade: Campus Russas				
1.2. Curso: Engenharia de Software				
1.3. Estrutura Curricular (ano-período): 2018.1				
1.4. Nome da Disciplina: Arquitetura de Software				
1.5. Código da Disciplina: RUS0242				
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: 64	C.H. Teórica: 64	C.H. Prática: ---	C.H. EaD: ---	C.H. Extensão: ---
1.9. Pré-requisitos (quando houver): Projeto Detalhado de Software				
1.10. Co-requisitos (quando houver): ---				
1.11. Equivalências (quando houver): Arquitetura de Software				
1.12. Professor(es): Anna Beatriz dos Santos Marques				
2. Justificativa				

A cada dia os sistemas de software se tornam parte integrante do cotidiano das pessoas. Esses sistemas são responsáveis por controlar desde simples ações, como abrir uma porta, até operações complexas como controlar aviões, gerenciar sistemas de telecomunicação e transações financeiras. Junto com o aumento do tamanho e da complexidade dos sistemas software, surge a necessidade de produzir sistemas cada vez mais confiáveis. Por outro lado, devido à competição de mercado, a necessidade das organizações por redução de tempo e custos no desenvolvimento e manutenção de software, aumenta. Nesse cenário, arquitetura de software tem assumido um papel importante no desenvolvimento de sistemas de software. Ela representa o núcleo de cada sistema de software bem projetado. A arquitetura de software lida com os “blocos de construção” de alto nível, que representam o sistema de software subjacente. Esses “blocos de construção” são os componentes (unidades de processamento em um sistema), os conectores (modelos das interações entre componentes de software), e as configurações (arranjos de componentes de software e conectores, e as regras que orientam sua composição). Essas abstrações proveem meios para que o engenheiro de software possa

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.

avaliar se o sistema em desenvolvimento está alinhado aos seus requisitos e aos interesses da organização. Desse modo, a adoção da arquitetura correta pode ajudar na gerência da complexidade e trazer diversos benefícios, tais como aumento da confiabilidade, manutenibilidade e redução de riscos. Dentro da matriz curricular do curso, a disciplina de Arquitetura de Software, além de apresentar novas técnicas e abordagens de abstração e modelagem de software sob uma perspectiva integrada e sistêmica, busca aprofundar conceitos básicos sobre modularidade, modelagem e análise de sistemas de software introduzidos nas disciplinas de “Programação Orientada a Objetos”, “Projeto Detalhado de Software” e “Análise e Modelagem de Software”, e introduzir conceitos de reuso de software alvo da disciplina de “Reuso de Software”. Além do aprofundamento conceitual, a disciplina prioriza atividades teórico/prático associadas à modelagem, documentação e análise de arquiteturas, compreensão do formato padrão utilizado por engenheiros de software para documentar arquiteturas de famílias de aplicações para uma futura reutilização e como implantar, de forma sistemática, uma arquitetura.

3. Ementa

Definição de arquitetura de software. A importância e o impacto em um empreendimento de software. Estilos arquiteturais (*pipes-and-filters*, *camadas*, *publish-subscribe*, baseado em eventos, cliente-servidor, dentre outros). Relação custo/benefício entre vários atributos arquitetônicos. Questões de hardware em projeto de software. Rastreabilidade de requisitos e arquitetura de software. Arquiteturas específicas de um domínio e linhas de produtos de software. Notações arquiteturais (ex., visões, representações e diagramas de componentes). Reutilização em nível arquitetural.

4. Objetivos – Geral e Específicos

Objetivos Gerais:

- Compreender o conceito de arquitetura de software e a sua importância para o sucesso de um empreendimento de software;
- **Compreender as dificuldades e a forma como projetar, analisar, documentar e implantar uma arquitetura software dentro de uma organização.**

Objetivos específicos:

- Documentar arquiteturas utilizando padrões e estilos arquitetônicos;
- Identificar aspectos de qualidade de software que restringem uma arquitetura de software;
- Utilizar métodos sistemáticos para análise de arquiteturas de software;
- Compreender normas técnicas referentes à arquitetura de software;
- Identificar e gerenciar conflitos que podem afetar a arquitetura de software;
- Avaliar a qualidade de uma arquitetura de software;
- **Mensurar os custos e os benefícios de uma arquitetura software para a organização.**

5. Calendário de Atividades

Data	Descrição do Conteúdo	Carga Horária
14/03/2023	Definição de arquitetura de software <i>Apresentação da ementa e cronograma de provas e trabalhos</i>	2h
16/03/2023	A importância e o impacto em um empreendimento de software e conceitos de arquitetura de software	2h
21/03/2023	Processo de design arquitetural	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>Sorteio de temas para o 1º seminário (tipos de arquitetura de software: LPS, MDA, ecossistemas de software, big data, SOA)</i>	
23/03/2023	Arquitetura e Requisitos.	2h
28/03/2023	Identificando requisitos arquiteturais.	2h
30/03/2023	Sem aula	-
04/04/2023	Sem aula	-
06/04/2023	Recesso escolar e Ponto facultativo – Semana Santa	-
11/04/2023	Atributos de qualidade.	2h
13/04/2023	Identificando requisitos arquiteturais.	2h
18/04/2023	SESCOMP - Semana de Engenharia de Software e Ciência da Computação	2h
20/04/2023	SESCOMP - Semana de Engenharia de Software e Ciência da Computação	2h
25/04/2023	Seminário sobre tipos de arquiteturas de software	2h
27/04/2023	Seminário sobre tipos de arquiteturas de software	2h
02/05/2023	Estilos arquiteturais e Padrões arquiteturais.	2h
04/05/2023	Estilos arquiteturais e Padrões arquiteturais.	2h
09/05/2023	Decisões arquiteturais	2h
11/05/2023	Experiência com jogo educacional LEARN (Learning software ARchitecture fuNdamentals)	2h
16/05/2023	Experiência com jogo educacional LEARN (Learning software ARchitecture fuNdamentals)	2h
18/05/2023	Prova Parcial 1 <i>Conteúdo: Conceitos de Arquitetura de Software; Design Arquitetural; Requisitos e Arquitetura. Padrões e Estilos Arquiteturais. Decisões Arquiteturais.</i>	2h
23/05/2023	Documentação de Arquiteturas de Software. Notações Arquiteturais. Visões Arquiteturais. <i>Explicação do Trabalho Prático - Definição de equipes e distribuição das especificações iniciais dos sistemas</i>	2h
25/05/2023	Modelagem arquitetural – Palestra Leonardo Barreto (USES – UFAM)	2h
30/05/2023	Levantamento de requisitos para o projeto da disciplina	2h
01/06/2023	Modelagem arquitetural com modelo 4+1	2h
06/06/2023	Modelagem arquitetural com modelo 4+1	2h
08/06/2023	Ponto facultativo – Corpus Christi	-
13/06/2023	Modelagem arquitetural com modelo 4+1	2h
15/06/2023	Modelagem arquitetural – exercícios em grupo	2h
20/06/2023	Modelagem arquitetural com abordagem C4	2h
22/06/2023	Modelagem arquitetural com abordagem C4	2h
27/06/2023	Modelagem arquitetural com abordagem C4	2h
29/06/2023	Modelagem arquitetural – exercícios em grupo	2h

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.

04/07/2023	Avaliação e Teste de Arquitetura de Software	2h
06/07/2023	Entrega do Trabalho Prático	2h
11/07/2023	Prova Parcial 2	2h
13/07/2023	Prova de 2ª chamada	2h
18/07/2023	Avaliação Final	-
20/07/2023	-	-
22/07/2023	Consolidação das sínteses de notas e frequências 2023.1 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 Arquitetura de Software; Aprendizagem baseada em projetos para a realização de atividades práticas do processo de design arquitetural em software; Seminário para apresentação de resultados de pesquisas e leituras/análises de artigos científicos sobre desafios e objetos de pesquisa atuais no contexto de Arquitetura de Software.	
7. Atividades Discentes	
Atividades práticas em sala de aula para o entendimento e seleção de estilos e padrões arquiteturais; modelagem arquitetural e revisões arquiteturais. Desenvolvimento de projetos para aplicar os conteúdos de Design Arquitetural com foco em Decisões Arquiteturais, Seleção de estilos e padrões arquiteturais e Descrição Arquitetural. Preparação de Seminários para apresentar os resultados obtidos no projeto e em leituras/análise de artigos científicos sobre Arquitetura de Software.	
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 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). As notas da disciplina são compostas de duas Avaliações Progressivas (AP1 e AP2). A Avaliação Progressiva 1 (AP1) é composta de: - BÔNUS Atividades Práticas 1 (ATP1): Atividades práticas sobre os conteúdos abordados (0,0 – 1,0) - Seminário 1 (SEM1): Apresentação sobre tipos de arquitetura de software (0,0 – 5,0) - Prova Parcial 1 (PP1): Prova escrita (0,0 – 5,0) A Avaliação Progressiva 2 (AP2) é composta de: - BÔNUS Atividades Práticas 2 (ATP2): Atividades práticas sobre os conteúdos abordados (0,0 – 1,0)	

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.

- Trabalho Prático 1 (TP1): Requisitos + Estilos e Padrões Arquiteturais + Modelagem Arquitetural (0,0 – 5,0)

- Prova Parcial 2 (PP2): Prova escrita (0,0 – 5,0)

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:

$$\text{NotaFinal} = (\text{MediaAPs} + \text{AF}) / 2$$

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 (sugere-se a inclusão de, pelo menos, 03 títulos):

1. TAYLOR, R. N.; MEDVIDOVIC, N.; DASHOFTY, E. M. **Software architecture: Foundations, Theory, and Practice**. Wiley, 2009. 750 p.
2. CLEMENTS, Paul et al. **Documenting software architectures: views and beyond**. 2. ed. Massachusetts: Addison-Wesley Professional, 2010. 592 p.
3. BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. **Software architecture in practice**. 2. ed. Boston, Massachusetts: Addison-Wesley Professional, 2003. 560 p.

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):

- SHAW, Mary; GARLAN, David. **Software architecture: perspectives on an emerging discipline**. São Paulo: Prentice Hall, 1996. 242 p.
- SOMMERVILLE, I. **Engenharia de software**. 7. ed. São Paulo: Pearson Addison-Wesley, 2007.
- GORTON, Ian. **Essential software architecture**. Berlin: Springer, 2006. 283 p. ISBN 3540287132 (enc.).
- REEKIE, John. **A software architecture primer**. Sydney, Australia: Angophora Press, 2006. 179 p. ISBN 0646458418 (broch.).
- BEZERRA, E. **Princípios de análise e projeto de sistemas com UML**. 2. ed. Rio de Janeiro: Elsevier, 2007.

10. Parecer

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.

Assinatura do Professor

____/____/____

Professor Responsável

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.