

Introdução e Funcionamento

ARQSOFT 2023-2024

Propósitos

Esta unidade curricular promove conhecimentos e competências de arquitetura de software, principalmente de aplicações empresariais distribuídas.

Especificamente, no final desta UC o estudante deve ser capaz de:

- CO1. Debater conceitos, princípios e processos (métodos e táticas) de arquitetura de software, e justificar a sua adequação em situações específicas (Nível de Bloom (NL): 5/6)
- CO2. Investigar e sistematizar vetores arquiteturais e fatores contextuais do software a desenvolver (NL: 5/6)
- CO3. Desenhar a arquitetura duma aplicação de software adotando estilos e padrões arquiteturais, e considerando vetores arquiteturais e fatores contextuais para chegar a um compromisso entre benefícios e riscos potenciais na avaliação da sua adequação (NL: 6/6)
- CO4. Construir a aplicação de software a partir do design arquitetural mais promissor (nesse momento), adotando revisões periódicas (NL: 6/6)
- CO5. Testar e avaliar a aplicação de software desenvolvida considerando os vetores e fatores contextuais definidos (NL: 5/6)

Conteúdos

P1. Arquitetura de software (10%)

- Atividades de ciclo de vida
- Stakeholders e papéis

P2. Drivers arquitetónicos (10%)

- Finalidade da conceção
- Atributos de qualidade
- Requisitos arquiteturais significativos (ASR)
- Funcionalidade(s) primária(s)
- Preocupações arquiteturais
- Restrições

P3. Conceitos de Design (20%)

- Estilos e arquiteturas de referência
- Padrões: exemplo(s) e catálogos
- Táticas: exemplo(s) e catálogos
- Componentes externos

P4. Conceção arquitetural com Attribute-Driven Design (ADD) (30%)

- ADD v3.0: principais características
- Etapas e natureza iterativa
- Decisões de conceção arquitetural como um processo
- Cost Benefit Analysis Method (CBAM)
- Documentação durante a conceção
- Outras atividades

P5. Microserviços (30%)

- Características
- Estratégias de migração e decomposição
- Padrões de conceção arquitetural para sistemas distribuídos
- Outros padrões e estilos de arquitetura distribuída

Bibliografia recomendada

- Bass, L., Clements, P., Kazman, R. (2021); Software Architecture in Practice (4th edition), Addison-Wesley. ISBN: 978-0-13-688609-9.
- Cervantes, H., Kazman, R. (2016). Designing software architectures: a practical approach. Addison-Wesley. ISBN: 978-013-4390789.
- Richards, M., Ford, N. (2020). Fundamentals of Software Architecture: An Engineering Approach. O'Reilly Media. ISBN: 978-1-492-04345-4.
- Pacheco, V. (2018). Microservice Patterns and Best Practices; Packt Publishing. ISBN: 1788474031.
- Newman, S. (2021). Building Microservices: Designing Fine-Grained Systems; O'Reiley. ISBN: 9781492033974.
- Martin, R. C. (2017); Clean Architecture: A Craftsman's Guide to Software Structure and Design, Prentice Hall. ISBN: 978-0134494166.
- Hohpe, G, Woolf, B. (2004). Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions. Addison-Wesley, ISBN: 0321200683.
- Snyder, B., Bosanac, D., Davies, R. (2011). ActiveMQ in Action. Manning Publications Co. ISBN: 978-1933988948.
- Ibsen, C., Anstey, J. (2010). Camel in Action. Manning Publications Co. ISBN: 9781935182368.

Abordagem pedagógica

- Iterativa e Incremental
- 1 tema/1 projeto/2 iterações
- Feedback frequente (Feedback*, Early Warning* e Embrace Correction*)
- Avaliação individual (Fair Project Grading*; Fair Team Grading*)

*) em Pedagogical Patterns: Advice for Educators; Pedagogical Pattern Editorial Board: Joseph Bergin, Jutta Eckstein, Markus Völter, Marianna Sipos, Eugene Wallingford, Klaus Marquardt, Jane Chandler, Helen Sharp, Mary Lynn Manns (Eds.); Published by Joseph Bergin Software Tools; 2012.

Critérios de avaliação



Instrumento	Período de realização	Peso na nota final			Nota mínima
Projeto 1	semanas 1 - 7	75%	100%	30%	9,5
	ressubmissão aquando de Projeto 2 (opcional)	25%	0		
Projeto 2	semanas 8- 15	40%			8,0
Exame		30%			

Melhoria de nota (apenas depois de aprovação):

- Exame: mesma fórmula de cálculo da nota
- Global (prova aplicada): correspondente a 100% da nota

Aulas

- Teóricas
 - Debate de abordagens
 - Debate de exemplos e casos de estudo
- Teórico-práticas
 - Workshop de eliminação de requisitos
 - Workshop de caracterização de requisitos e cenários arquiteturais
 - Workshop de design arquitetural
- Laboratoriais
 - Proof-of-concept de abordagens arquiteturais
 - Desenvolvimento de software de acordo com o design selecionada
 - Retorno e avaliação sobre projeto (análise, design, implementação)

Funcionamento

- As aulas são fundamentais:
 - nem todo o conteúdo da UC é disponibilizado nos documentos ou apresentações publicadas
 - nem todas as competências a adquirir na UC podem ser adquiridas na bibliografia
- Trabalho autónomo é fundamental:
 - nem todo o conteúdo da UC é disponibilizado nos documentos ou apresentações publicadas
 - nem todas as competências a adquirir na UC podem ser adquiridas na bibliografia e nas aulas