



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

EMENTA
Arquitetura de sistemas computacionais, funções do arquiteto de sistemas, divisão de sistemas em camadas, micros serviços, bibliotecas e APIs, sistemas distribuídos, sistemas multiplataforma, bancos de dados distribuídos, webservices e afins.

SYLLABUS
Computational systems architecture, systems architect functions, systems division into layers, microservices, libraries and APIs, distributed systems, multiplatform systems, distributed databases, webservices and the like.

TEMÁRIO
Arquitectura de sistemas informáticos, funciones de arquitecto de sistemas, división de sistemas en capas, microservicios, librerías y APIs, sistemas distribuidos, sistemas multiplataforma, bases de datos distribuidas, servicios web y similares.

PROFESSORES QUE LECIONAM A DISCIPLINA
<ul style="list-style-type: none"> <li>RODRIGO BOSSINI TAVARES MOREIRA</li> <li>ANTONIO SERGIO FERREIRA BONATO</li> </ul>

ESTRATÉGIAS ATIVAS PARA APRENDIZAGEM - EAA
Aulas de Teoria? - <b>Não</b>
Aulas de Exercício? - <b>Não</b>
Aulas de Laboratório? - <b>Sim</b>

MODALIDADE DE ENSINO
- Aula de Laboratório - Presencial



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

ATIVIDADES DE EXTENSÃO
<p>A disciplina proporciona interação dialógica (ação de mão dupla) dos estudantes com a comunidade externa de forma ativa e prática? <b>Não</b></p> <p>Modalidades:</p> <p>Porcentagem:</p> <p>Atividades e Contribuições:</p>

OBJETIVOS - Conhecimentos, Habilidades, e Atitudes
<p>1. Conhecimentos 1.1 Familiarizar-se com os conceitos de arquitetura de software e padrões de projeto. 1.2 Entender os diferentes caminhos de formação do arquiteto de software. 1.3 Entender as diferentes funções do arquiteto de software. 1.4 Entender os principais padrões arquiteturais. 1.5 Entender o funcionamento dos principais padrões de projeto estruturais, criacionais e comportamentais. 1.6 Entender vantagens e desvantagens referentes ao uso de diferentes padrões arquiteturais. 1.7 Conhecer as principais bibliotecas e frameworks para o desenvolvimento de software utilizando micro serviços. 1.8 Entender o funcionamento de sistemas e bases de dados distribuídas. 1.9 Entender processos de refatoração de soluções de software. 2. Habilidades 2.1 Implementar soluções de software utilizando padrões arquiteturais e padrões de projeto. 2.2 Refatorar soluções de software monolíticas para que utilizem micro serviços. 2.3 Refatorar soluções de software para que utilizem padrões arquiteturais e padrões de projeto. 2.4 Implementar soluções de software baseadas em micro serviços utilizando bibliotecas e frameworks atuais. 3. Atitudes 3.1. Ser proativo na busca de soluções. 3.2. Trabalhar em grupo. 3.3 Tomar decisões de forma autônoma ou em grupo, de acordo com o contexto. 3.4 Exercer papel de liderança e condução de trabalho.</p>



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

COMPETÊNCIAS DESENVOLVIDAS
<p><b>Competência1:</b> Atuar na tomada de decisão estratégica em projetos de desenvolvimento e manutenção de sistemas.</p> <p><b>Competência2:</b> Indicar padrões arquiteturais e de projeto apropriados levando em conta diferentes restrições inerentes ao contexto.</p> <p><b>Competência3:</b> Aplicar padrões de arquitetura e padrões de projeto adequados às necessidades do projeto e aos padrões adotados pela organização.</p>

METODOLOGIA DIDÁTICA
As aulas terão caráter prático, sendo ministradas em laboratório. O professor fará uso de microcomputador e de recursos audiovisuais acoplados a ele. Os alunos terão microcomputadores disponíveis individualmente. Os trabalhos práticos serão caracterizados pelo desenvolvimento de pequenas soluções de software empregando os diferentes conceitos abordados ao longo do período letivo.

INSTRUMENTOS DE AVALIAÇÃO
<p><b>2. Instrumentos:</b></p> <ul style="list-style-type: none"> <li>- Provas</li> <li>- Trabalhos - Individual e/ou em Equipe</li> </ul>

CONHECIMENTOS PRÉVIOS NECESSÁRIOS PARA O ACOMPANHAMENTO DA DISCIPLINA
1. Desenvolvimento de software utilizando ao menos uma linguagem de programação com suporte à orientação a objetos (C++, Java, C# etc). 2. Noções de redes de computadores e protocolos.

CONTRIBUIÇÃO DA DISCIPLINA
A disciplina aborda aspectos sobre o papel do arquiteto de software e sobre padrões arquiteturais e de projeto, além do uso de frameworks e bibliotecas modernas, o que é fundamental para a formação do Engenheiro de Computação.



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

TEXTO PARA O SITE
<p>O desenvolvimento de software de alta qualidade requer a escolha de padrões arquiteturais apropriados que deve ser feita considerando-se suas principais vantagens e desvantagens. Tal atividade deve ser desempenhada por profissional altamente qualificado que, em geral, recebe o nome de arquiteto de software. Além de conhecer os diferentes padrões arquiteturais e de projeto, o arquiteto de software precisa conhecer frameworks e bibliotecas atuais, já que esse conhecimento técnico viabiliza escolhas mais acertadas. Nesta disciplina são abordados:- conceitos fundamentais sobre o papel do arquiteto de software- os principais padrões arquiteturais e de projeto e suas vantagens e desvantagens- técnicas de refatoração de código- desenvolvimento prático de software utilizando bibliotecas e frameworks atuais.</p>
BIBLIOGRAFIA BÁSICA
<p><b>Bibliografia Básica:</b></p> <p>COULOURIS, George DOLLIMORE, Jean KINDBERG, Tim et al. Sistemas Distribuídos. Porto Alegre: Bookman, 2013. E-book.</p> <p>FOWLER, Martin. Padrões de Arquitetura de Aplicações Corporativas. Porto Alegre: Bookman, 2006. E-book.</p> <p>GAMMA, Erich HELM, Richard JOHNSON, Ralph et al. Padrões de projetos: soluções reutilizáveis de software orientados a objetos. Porto Alegre: Bookman, 2000. E-book</p> <p>PEREIRA, Mariana Araújo NEUMANN, Fabiano Berlinck MILANI, Alessandra M. Paz et al. Framework de Big Data. Porto Alegre: SAGAH, 2020. E-book.</p> <p>PRESSMAN, Roger S MAXIM, Bruce R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016. 940 p.</p> <p>- Richards, M. Fundamentals of Software Architecture: An Engineering Approach. 1st edition. O'Reilly Media. 2020. 432 p. ISBN-10 1492043451. ISBN-13 978-1492043454.</p>



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

BIBLIOGRAFIA COMPLEMENTAR
<p><b>Bibliografia Complementar:</b></p> <p>BERSON, Alex. Client/server architecture. New York: McGraw-Hill, 1992. 452 p.</p> <p>COMER, Douglas E STEVENS, David L. Internetworking with TCP/IP. Englewood Cliffs, NJ: Prentice-Hall, 2001. v. 3. 601 p.</p> <p>FLANAGAN, David. JavaScript: o guia definitivo. Porto Alegre: Bookman, 2013. E-book.</p> <p>GROOVER, Mark et al. Hadoop application architectures. 1. ed. Sebastopol, CA: O'Reilly, c2015. 373 p.</p> <p>HORSTMANN, Cay S CORNELL, Gary. Core Java. SCHAFRANSKI, Carlos (Trad.), FURMANKIEWICZ, Edson (Trad.). 8. ed. São Paulo: Pearson, 2010. v. 1. 383 p.</p> <p>OLIVEIRA, Cláudio Luís Vieira ZANETTI, Humberto Augusto Piovesana. Node.js: programe de forma rápida e prática. São Paulo: Expressa, 2021. E-book.</p>

## AVALIAÇÃO (Conforme Resolução RN CEPE 16/2024) e CRITÉRIOS DE APROVAÇÃO

**Critério de aprovação: C2/2007****Quantidade de Trabalhos: 2**

- K1:5
- K2:5

Peso de MP(kp): 5

Peso de MT(kt): 5

## INFORMAÇÕES SOBRE PROVAS E TRABALHOS

Trabalho 1: Desenvolvimento e apresentação da primeira fase do projeto em grupo. Trabalho 2: Desenvolvimento e apresentação da segunda fase do projeto em grupo. Cada trabalho tem peso igual a 0,5. Prova 1: Avaliação escrita. Peso igual a 0,2. Prova 2: Avaliação escrita. Peso igual a 0,2. Prova 3: Avaliação escrita. Peso igual a 0,3. Prova 4: Avaliação escrita. Peso igual a 0,3.

## OUTRAS INFORMAÇÕES



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

SOFTWARES NECESSÁRIOS PARA A DISCIPLINA
1. Visual Studio Code ( <a href="https://code.visualstudio.com/">https://code.visualstudio.com/</a> ) 2. JDK versão 11 ( <a href="https://www.oracle.com/java/technologies/javase-jdk11-downloads.html">https://www.oracle.com/java/technologies/javase-jdk11-downloads.html</a> ) 3. Node Version Manager ( <a href="https://github.com/coreybutler/nvm-windows">https://github.com/coreybutler/nvm-windows</a> ) 4. Node e NPM (Instalar usando o NVM, do item 3) 5. Python 3.x ( <a href="https://www.anaconda.com/distribution/">https://www.anaconda.com/distribution/</a> ) 6. AWS CLI ( <a href="https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html">https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html</a> ) 7. Postman ( <a href="https://www.postman.com/">https://www.postman.com/</a> ) 8. MySQL Community e Workbench: ( <a href="https://dev.mysql.com/downloads/">https://dev.mysql.com/downloads/</a> ) 9. Eclipse Enterprise ( <a href="https://www.eclipse.org/downloads/packages/release/2020-12/r/eclipse-ide-enterprise-java-developers">https://www.eclipse.org/downloads/packages/release/2020-12/r/eclipse-ide-enterprise-java-developers</a> ) 10. Apache Tomcat 9 ( <a href="https://tomcat.apache.org/download-90.cgi">https://tomcat.apache.org/download-90.cgi</a> ) 11. Docker ( <a href="https://www.docker.com/">https://www.docker.com/</a> ) 12. Hadoop ( <a href="https://hadoop.apache.org">https://hadoop.apache.org</a> )

PROGRAMA DA DISCIPLINA		
Semana	Conteúdo	EAA
1L	Atividades de Planejamento e Capacitação Docente	0;
2L	Programa de Recepção e Integração dos Calouros (PRINT).	0;
3L	Apresentação do plano de ensino. Introdução à arquitetura de software e a padrões de projeto.	1% a 10%;
4L	Caminhos de formação do arquiteto de software e os seus papeis na indústria de software.	1% a 10%;
5L	Arquitetura cliente/servidor. Protocolo HTTP	1% a 10%;
6L	Padrão arquitetural REST. Web services REST.	1% a 10%;
7L	Modelo de maturidade de Richardson.	1% a 10%;
8L	Uso apropriado dos métodos do protocolo HTTP.	11% a 40%;
9L	Web services RESTful com NodeJS.	11% a 40%;
10L	Prova 1.	11% a 40%;
11L	Feriado.	11% a 40%;
12L	Modularidade. Alta coesão e baixo acoplamento.	11% a 40%;
13L	Feriado.	11% a 40%;
14L	Padrões de projeto criacionais.	11% a 40%;
15L	Padrões de projeto estruturais.	11% a 40%;
16L	Padrões de projeto comportamentais.	11% a 40%;
17L	Arquiteturas monolíticas e distribuídas.	41% a 60%;
18L	Introdução a micro serviços.	41% a 60%;
19L	Implementação de micro serviços.	41% a 60%;
20L	Prova 2.	41% a 60%;
21L	Atividades de Planejamento e Capacitação Docente	41% a 60%;



## PLANO DE ENSINO

IDENTIFICAÇÃO		
Disciplina: ARQUITETURA DE SISTEMAS COMPUTACIONAIS		Código da Disciplina: ECM516
Course: System Architecture		
Materia: Arquitetura de Sistemas Computacionais		
Periodicidade:	Carga Horária total: 80	
Professor Responsável: ANTONIO SERGIO FERREIRA BONATO		Titulação MESTRADO

## PLANO DE ENSINO PARA O ANO LETIVO DE 2025

22L	Prova SUB1.	41% a 60%;
23L	Implementação de micro serviços.	41% a 60%;
24L	Refatoração de arquiteturas monolíticas. Containers Docker.	41% a 60%;
25L	Implantação de micro serviços com Docker.	41% a 60%;
26L	Orquestração de contêineres.	61% a 90%;
27L	Introdução ao Kubernetes.	61% a 90%;
28L	Arquitetura do Kubernetes e seu uso com kubectl.	61% a 90%;
29L	Pods, Deployments e Réplicas. Implantação do Kubernetes em serviços de computação em nuvem.	61% a 90%;
30L	Prova 3.	61% a 90%;
31L	Introdução a sistemas distribuídos.	61% a 90%;
32L	Arquiteturas paralelas e distribuídas.	91% a 100%;
33L	Modelos de comunicação em sistemas distribuídos.	61% a 90%;
34L	Algoritmos para exclusão mútua em sistemas distribuídos.	61% a 90%;
35L	MapReduce e Spark.	91% a 100%;
36L	Sistemas Gerenciados de Bancos de Dados Distribuídos(SGBDD). Arquiteturas de SGBDDs.	91% a 100%;
37L	Feriado.	91% a 100%;
38L	Prova 4.	91% a 100%;
39L	Big Data e NoSQL.	91% a 100%;
40L	Prova SUB2.	91% a 100%;
41L	Revisão.	91% a 100%;
42L	Revisão.	91% a 100%;