

Apresentação da Disciplina

QXD0099 - Desenvolvimento de Software para Persistência

Universidade Federal do Ceará - *Campus* Quixadá

Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br



Agenda

- Apresentações
- Disciplina
 - Ementa
 - Objetivos gerais e específicos
 - Metodologia de ensino
 - Conteúdo programático
- Formato de avaliação
- Formas de aprovação e reprovação
- Faltas
- Repositório e canal de comunicação
- Bibliografia
- Códigos úteis

Apresentações

Professor, alunos e expectativas

Apresentações



Prof. Victor Pinheiro

- Graduação em Sistemas de Informação - UFC Quixadá (2017-2021)
- Mestrado em Computação - UFC Quixadá (2021-2023)
- Doutorado em Andamento em Computação - MDCC/UFC (2023-atual)
- Pesquisador do GREat/UFC

- **Áreas de interesse:** Ecossistemas de Software, Desenvolvimento de Software, Computação móvel e Ubíqua, Internet das Coisas, Informática aplicada na Saúde e Ciência de dados.
- **Lattes:** <http://lattes.cnpq.br/3822537365616539>
- **LinkedIn:** <https://www.linkedin.com/in/francisco-victor-da-silva-pinheiro-0b439214b/>

Apresentações

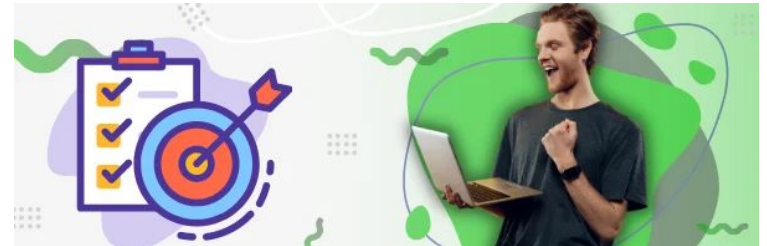
- **Alunos**

- Nome, Curso, Semestre
- De onde você vem?
- Para onde você vai? e
- Áreas de interesse



- **Sobre a Disciplina**

- *Quais as Expectativas?*



Disciplina

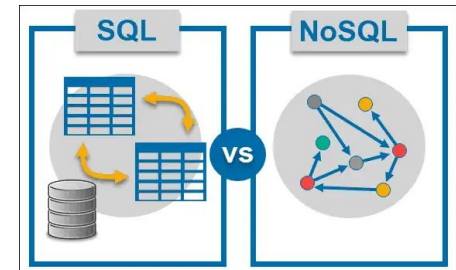
Ementa, objetivos gerais, específicos, metodologia de ensino e conteúdo programático

Disciplina

- Desenvolvimento de Software para Persistência
- Código: QXD0099
- Turma: 1A
- Turno: Tarde
- Carga Horária: 64h
- Local: Bloco 3 Laboratório 7
- Horários: QUA 15:30-17:30 | SEX 13:30-15:30

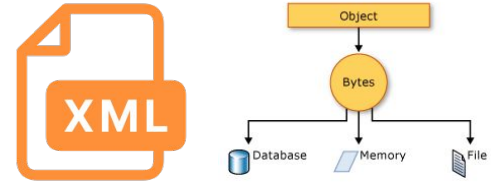
Disciplina

- Contextualização
 - Persistência é uma das questões fundamentais que afetam a arquitetura, projeto, implementação e implantação de sistemas de software.
 - O domínio dos conceitos e tecnologias de persistência de dados é importante para profissionais atuando com analistas, arquitetos, projetistas e desenvolvedores de software.
 - Os profissionais terão que tomar decisões técnicas na seleção e adaptação dos mecanismos de persistência mais adequados ao contexto do projeto, baseado no conhecimento e entendimento dos vários tipos de modelos disponíveis.



Disciplina

- Ementa
 - Definição de persistência.
 - Persistência empregando documentos XML, objetos serializáveis, SGBDs.
 - Tecnologias para persistência de informações.
 - Persistência de objetos usando base relacional. Persistência usando outros modelos de dados (orientado a documento, chave-valor, orientado a coluna, grafo).
- Importância para você e para o Curso.



Disciplina

- Objetivos Gerais
 - Identificar soluções de persistências adequadas às necessidades de stakeholders e contexto tecnológico;
 - Desenvolver componentes de software voltados para persistência, usando os principais modelos de dados existentes.



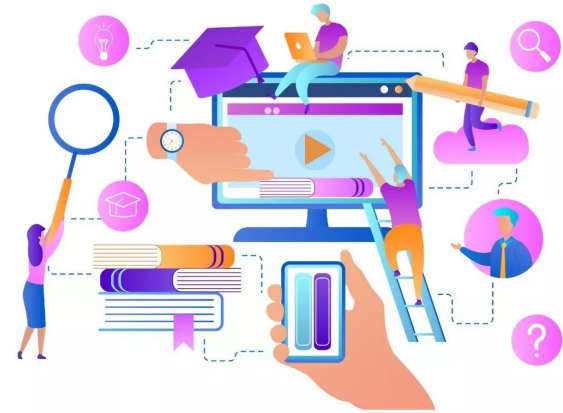
Disciplina

- Objetivos Específicos
 - Analisar o impacto de requisitos não-funcionais nas decisões de persistência;
 - Selecionar modelos de dados adequados para as funcionalidades dos sistemas;
 - Desenvolver software usando paradigma de orientação a objeto e banco de dados relacional;
 - Projetar, manipular e recuperar informações de banco de dados semi-estruturados;
 - Projetar, manipular e recuperar informações de banco de dados não-relacional, usando ferramentas de programação.



Disciplina

- Metodologia de ensino
 - Aprender fazendo
 - Desenvolvimento de:
 - Iniciativa ≠ Passividade
 - Autonomia
 - Auto-disciplina
 - Avaliação contínua
 - Trabalho em equipe
 - Cooperação / Ajuda mútua
 - Organização
 - Comunicação
 - Desinibição
 - Leitura, interpretação e produção de textos.
 - Responsabilidade
 - Competências específicas de cada assunto



Conteúdo Programático

- **Módulo 1: Arquivos em formatos texto e binário**
 - Persistência de Arquivos: texto, binário, CSV, Arquivos de propriedades Java.
 - Dados estruturados, semi-estruturados e não estruturados.
 - Serialização de objetos.
 - Arquivos em formato texto: XML. YAML. JSON, Arquivos de Logs.
 - Biblioteca Jackson
 - APIs de Logging
 - Arquivos em formato binário:
 - Protocol Buffer.
 - Compressão e Descompressão de arquivos (zip, gzip, rar, tar.gz, tar.bz).
 - Encriptação e Decriptação de arquivos (Simétrica e Assimétrica).
 - Verificação de integridade de arquivos (MD5, SHA-1, SHA-256, CheckSum).
 - Formato Parquet.

Conteúdo Programático

- **Módulo 2: Modelo relacional**
 - Modelo Relacional.
 - PostgreSQL e SQLite
 - Mapeamento Objeto-Relacional.
 - SQLAlchemy
 - SQLAlchemyModel
 - Migrações de Banco de Dados com Alembic
- **Módulo 3: Bancos não relacionais**
 - Bancos não relacionais, NoSQL e Big Data
 - Orientados à documentos - MongoDB
 - Orientados à colunas - Cassandra
 - Chave-valor - Redis
 - Database e backend como serviço - Firebase
 - API com Dados abertos
 - Object Storage – AWS S3 / MinIO
 - Recuperação de Informação / Busca - Elastic Search

Avaliação

Formato de avaliação, formas de aprovação, reprovação e faltas

Formato de Avaliação

- A avaliação da disciplina consiste de:
 - Trabalhos Práticos (**um por módulo em dupla**):
 - correspondendo a **50% da nota da disciplina**.
 - Trabalho Prático Final (**em equipe de 5 pessoas**):
 - Correspondendo a **30% da nota da disciplina**.
 - Criação de uma aplicação relacionada a dados abertos.
 - A aplicação deverá obter dados de fontes de dados abertas e correlacionar seus dados.
 - Resolução individual:
 - Listas de Exercícios relacionadas aos conteúdos ministrados em sala de aula e correspondendo a **20% da nota da disciplina**.
- **OBS: As Entregas ocorrerão mediante apresentação e envio do trabalho ao professor necessariamente realizada através do Moodle.**

Formas de aprovação e reprovação

- Nota Final = $(\text{Média dos Trabalhos Práticos} * 0.50) + (\text{Nota do Trabalho Prático Final} * 0.30) + (\text{Média das Listas de Exercícios} * 0.20)$
- **Onde:**
 - A média dos Trabalhos Práticos contribui com 50% da nota.
 - O Trabalho Prático Final contribui com 30% da nota.
 - As Listas de Exercícios contribuem com 20% da nota.

Formas de aprovação e reprovação

- Os critérios de aprovação são:
- Nota Final ≥ 7 : **Aprovado**.
- $4 \leq \text{Nota Final} < 7$: **Avaliação Final (AF)**.
- Nota Final < 4 : **Reprovado**.
- Na hipótese de você ir para a AF, deverá obter nota igual ou superior a 5,0 na avaliação final que somada à média das APs deverá resultar numa média igual ou superior a 5,0.

Faltas

- Necessita de 75% de presença em aula + nota para a aprovação.
- Mais de 25% de faltas causa **Reprovação por falta** na disciplina.
 - Reprovação por falta afeta muito o IRA
 - Prejudica programas acadêmicos como: AMOR, BIA e outras bolsas e programas acadêmicos.
 - Ou seja, prejudica muito o aluno, então **evite reprovar por falta**.

Comunicação

Repositório e canal de comunicação

Repositório e Canal de comunicação

- Repositório de códigos da disciplina:
 - https://github.com/pinheirovictor/2025.1_QXD009_9_persistencia-01A
- Linguagem de programação
 - Python + FastAPI
- Canal oficial de comunicação para avisos e notícias:
 - **SIGAA UFC**
- Outros canais de comunicação para dúvidas:
 - Email: victorpinheiro@ufc.br
 - Telegram: [@victorp07](https://t.me/victorp07)



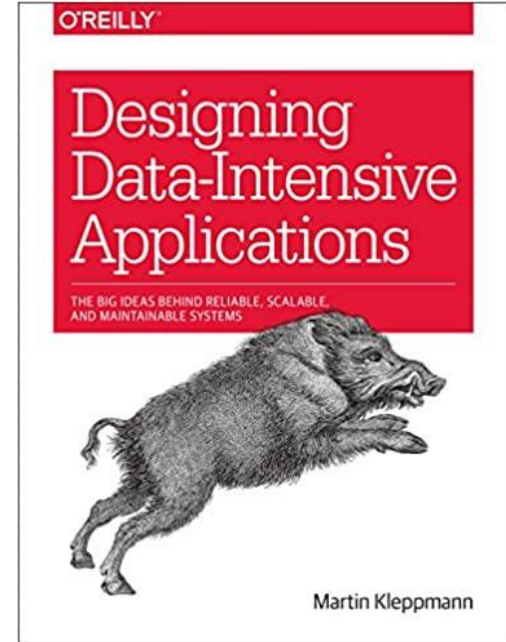
Bibliografia

Bibliografia básica, complementar e códigos úteis

Bibliografia

Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems

April 18, 2017
by Martin Kleppmann



Bibliografia

Part I. Foundations of Data Systems

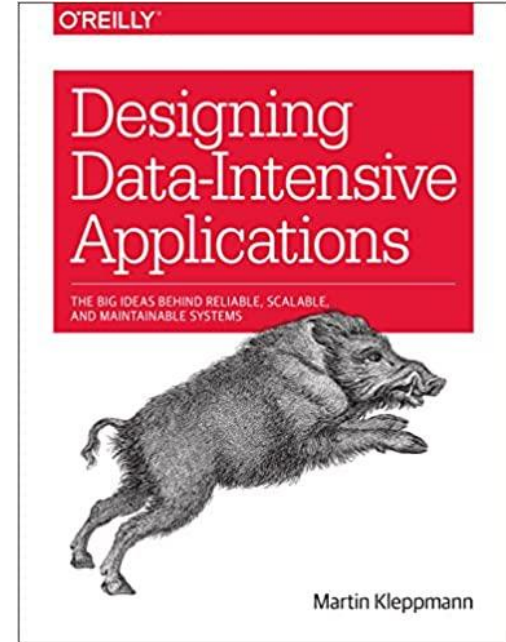
1. Reliable, Scalable, and Maintainable Applications
2. Data Models and Query Languages
3. Storage and Retrieval
4. Encoding and Evolution

Part II. Distributed Data

5. Replication
6. Partitioning
7. Transactions
8. The Trouble with Distributed Systems
9. Consistency and Consensus

Part III. Derived Data

10. Batch Processing
11. Stream Processing
12. The Future of Data Systems



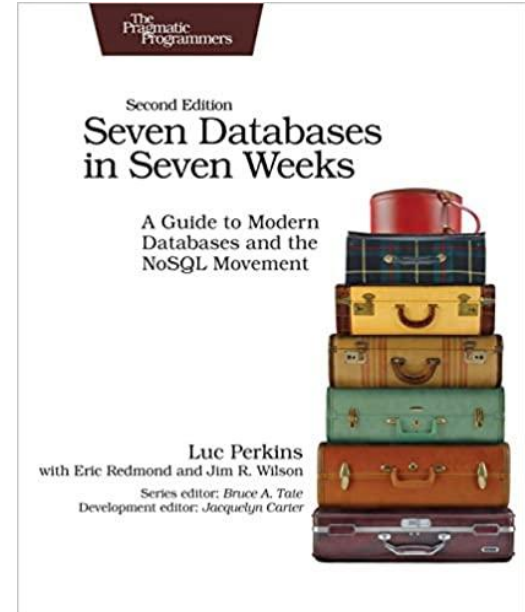
Bibliografia

Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement - 2nd Edition

April 24, 2018

Redis, Neo4J, CouchDB, MongoDB,
HBase, Postgres and DynamoDB

by Luc Perkins, Eric Redmond, Jim Wilson.

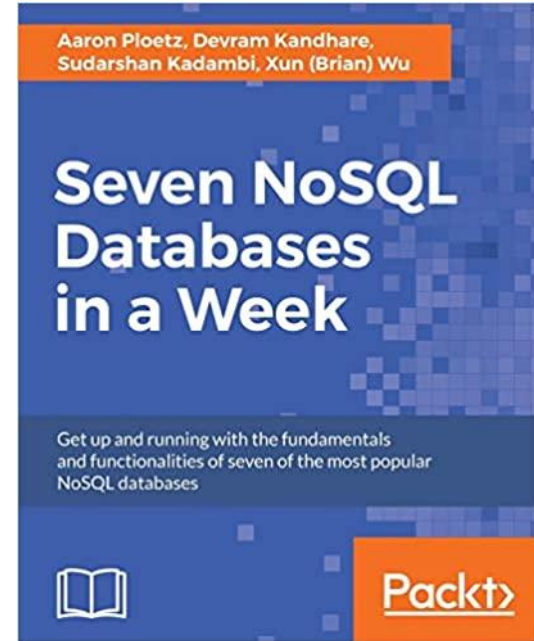


Bibliografia

Seven NoSQL Databases in a Week:
Get up and running with the fundamentals
and functionalities of seven of the most
popular NoSQL databases

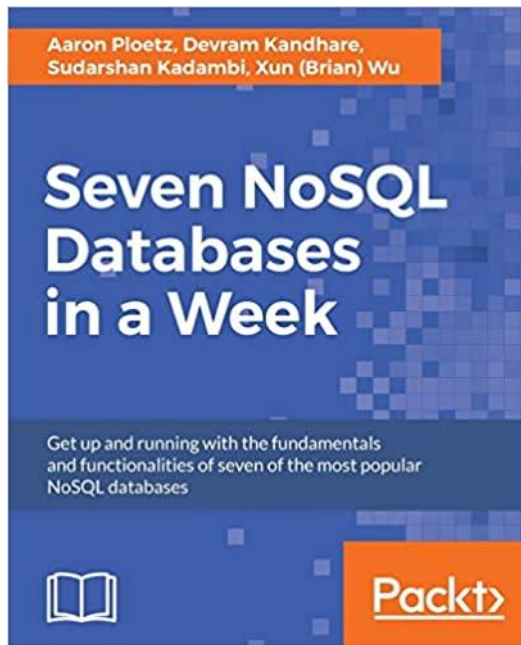
March 29, 2018

by Aaron Ploetz , Devram Kandhare ,
Sudarshan Kadambi, Xun (Brian) Wu



Bibliografia

1. Introduction to NoSQL Databases
2. MongoDB
3. Neo4j
4. Redis
5. Cassandra
6. HBase
7. Dynamo DB
8. InfluxDB



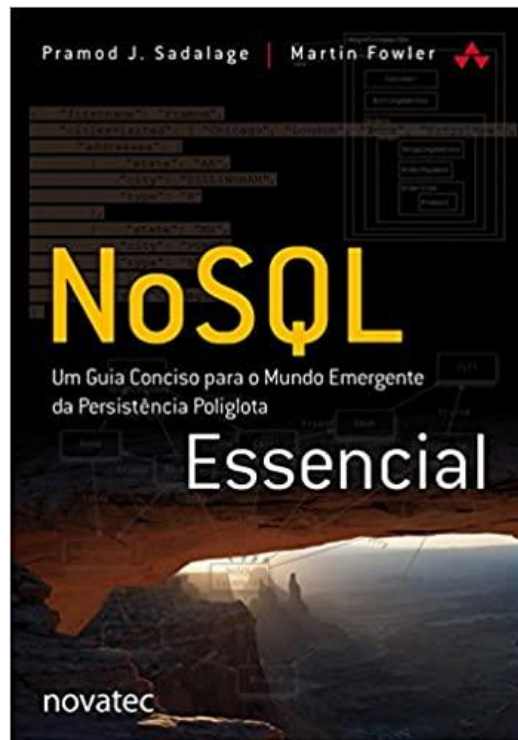
Bibliografia

NoSQL Essencial:

Um Guia Conciso Para o Mundo
Emergente da Persistência Poliglota

Junho 2013

por Martin Fowler, Pramod J. Sadalage



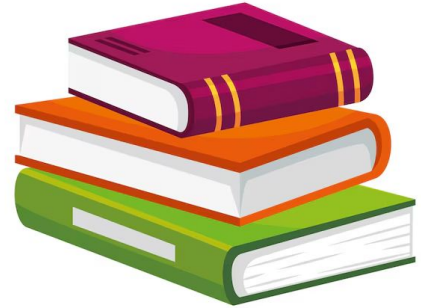
Bibliografia Básica

- SADALAGE, P. J. E FOWLER, M. NoSQL Essencial. Editora Novatec, São Paulo, 2013.
- REDMOND, E.; WILSON, J. R. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. 1ª edição, 2012. The Pragmatic Programmers.
- ULLMAN, J.D.; WIDOW, J. First Course in Database Systems. 3a edição, 2007. Prentice Hall.
- HAMBRICK, G. et al. Persistence in the Enterprise: A Guide to Persistence Technologies; 1ª edição, 2008. IBM Press.
- ELMASRI, R.; NAVATHE, S. B. Sistemas de banco de dados. 4ª edição, 2009. Pearson/Addison-Wesley.



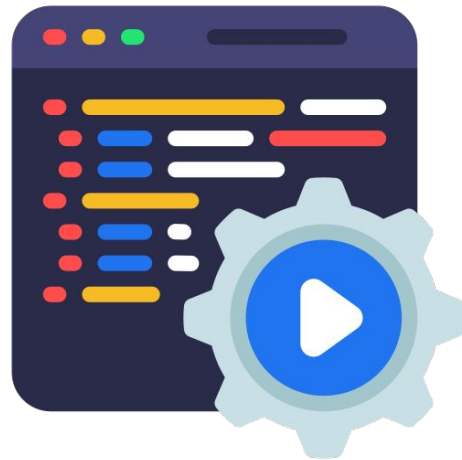
Bibliografia Complementar

- WHITE, Tom. Hadoop: the definitive guide. California: O'Reilly, 2009. xix, 501 p. ISBN 9780596521974 (broch.).
- AMBLER, S.W., SADALAGE, P.J. Refactoring Databases: Evolutionary Database Design. 1a edição, 2011. Addison Wesley.
- SILBERSCHATZ, A.; SUDARSHAN, S. Sistema de banco de dados. 2006. Campus.
- LYNN, B. Use a cabeça! SQL. 1ª edição, 2008. ALTA BOOKS.
- SMITH, Ben. JSON básico: conheça o formato de dados preferido da web. São Paulo: Novatec, 2015. 400 p. ISBN 9788575224366 (broch.).
- HITZLER, P., KRÖTZSCH, M., and RUDOLPH, S. (2009). Foundations of Semantic Web Technologies. Chapman & Hall/CRC.
- ANTONIOU, G. and HARMELEN, F. (2008). A Semantic Web Primer. Second Edition, Cambridge, MIT Press, Massachusetts.
- HEATH, T. and BIZER, C. (2011). Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool, 1st edition.



Códigos úteis

- [Git do professor Regis Pires Magalhães](#)
- [EntityManagerFilter](#)
- [JPAUtil](#)
- [GenericDAO](#)
- [GenericJPADAO](#) e uma [versão mais completa do GenericJPADAO](#)



Obrigado!

Dúvidas?



Universidade Federal do Ceará - *Campus* Quixadá

Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br

