



Pós-Graduação em Computação Distribuída e Ubíqua

INF628 - Engenharia de Software para Sistemas Distribuídos

Sandro S. Andrade
sandroandrade@ifba.edu.br

Objetivos



- Apresentar os principais desafios de engenharia durante o projeto e desenvolvimento de sistemas distribuídos modernos
- Apresentar e discutir as principais tecnologias, metodologias e processos de desenvolvimento atualmente utilizados na construção de sistemas distribuídos
- Proporcionar uma vivência prática acerca destas tecnologias

Roteiro do Curso



- Dia 1:
 - Principais desafios no desenvolvimento de sistemas distribuídos modernos
 - Padrões de Projeto para Sistemas Distribuídos
 - Eventos: Reactor, Proactor, Acceptor-Connector
 - Concorrência: Active Object, Leader-Followers
 - Estudo de caso: servidor web JAWS

Roteiro do Curso



- Dia 2:
 - Padrões de Projeto para Sistemas Distribuídos (cont.)
 - Concorrência: Half-Sync/Half-Async, Thread-Specific Storage
 - Acesso a serviço e configuração: Component Configurator, Interceptor, Extension Interface
 - Estudo de caso: servidor web Apache httpd
 - Soluções de middleware e frameworks para desenvolvimento de Sistemas Distribuídos
 - ACE, Twisted, Node.js, Boost.ASIO, cpp-netlib, pocoproject, icecpp

Roteiro do Curso



- Dia 3:
 - O Estilo Arquitetural Map-Reduce
 - Padrões de Projeto para Map-Reduce:
 - Numerical Summarizations, Inverted Index Summarizations, Counting with Counters, Filtering, Top Ten
 - O Hadoop

Roteiro do Curso



- Dia 4:
 - Padrões de Projeto para Map-Reduce (cont.):
 - Structured to Hierarchical
 - Reduce Side Join
 - Replicated Join
 - Composite Join

Roteiro do Curso



- Dia 5:
 - Arquiteturas de Referência para Cloud Computing:
 - IEEE-SA (P2301 / P2302)
 - ITU Cloud Computing and Standardization
 - NIST Special Publication 500-293
 - IBM Cloud Computing Reference Architecture
 - Rackspace Cloud: Private Edition Reference Architecture
 - AWS Architecture Center
 - Tecnologias para Cloud Computing:
 - OpenStack, Eucalyptus, Apache CloudStack, OpenNebula, OwnCloud, Rackspace, CloudFoundry, CloudXen, EngineYard

Roteiro do Curso



- Dia 6:
 - O Estilo Arquitetural REST
 - Variações do REST:
 - Baseado em consenso: A+REST, R+REST e REST+D
 - Livre de consenso: REST+E e ARRESTED
 - Computational REST (CREST)
 - O Estilo Arquitetural Tiles
 - Arquiteturas de Alta Disponibilidade

Roteiro do Curso



- Dia 7:
 - Apresentação do trabalho prático

Roteiro do Curso



- Dia 8:
 - Aula de dúvidas / revisão (18:30 – 19:30)
 - Avaliação escrita individual (19:30 – 22:00)

Bibliografia



- Livros:

- Pattern-Oriented Software Architecture Volume 2: Patterns for Concurrent and Networked Objects. Douglas Schmidt, Michael Stal, Hans Rohnert and Frank Buschmann. Wiley; Volume 2 edition. 2000. ISBN-13: 978-0471606956
- Pattern-Oriented Software Architecture Volume 4: A Pattern Language for Distributed Computing. Frank Buschmann, Kevlin Henney and Douglas C. Schmidt. Wiley; 1 edition. 2007. ISBN-13: 978-0470059029

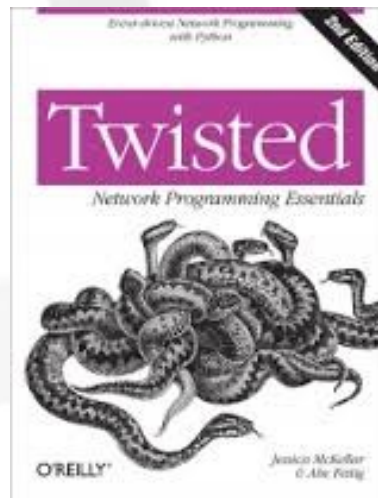
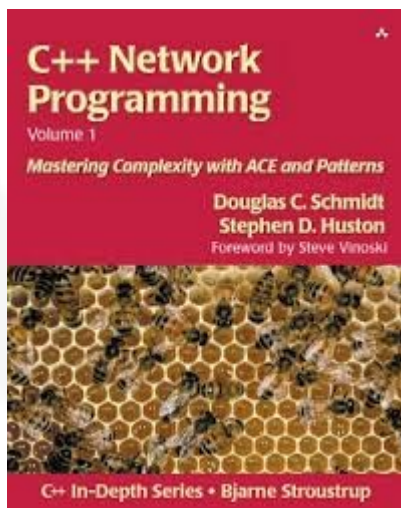


Bibliografia



- Livros:

- C++ Network Programming, Volume I: Mastering Complexity with ACE and Patterns. Douglas C. Schmidt and Stephen D. Huston. Addison-Wesley Professional. 2001. ISBN-13: 978-0201604641
- Twisted Network Programming Essentials. Jessica McKellar and Abe Fettig. O'Reilly Media; Second Edition edition. 2013. ISBN-13: 978-1449326111

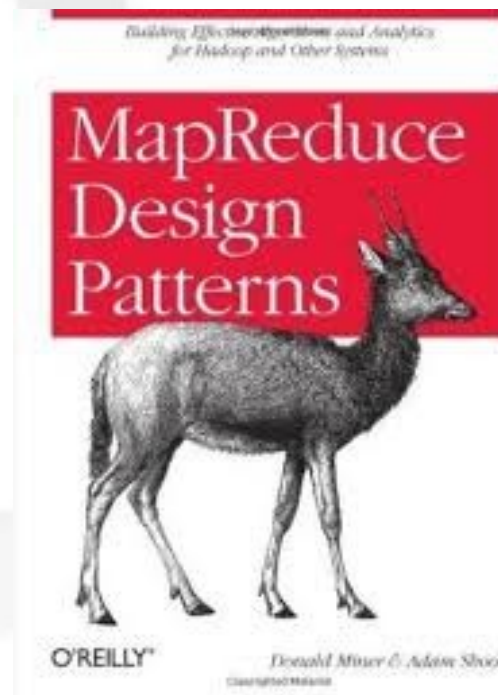


Bibliografia



- Livros:

- MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems. Donald Miner and Adam Shook. O'Reilly Media. 2012. ISBN-13: 978-1449327170



Bibliografia



- Artigos:

- JAWS: A Framework for High performance Web Servers. James C. Hu and Douglas C. Schmidt. 1999.
- Extending the RE presentational State Transfer (REST) Architectural Style for Decentralized Systems. Rohit Khare and Richard N. Taylor. ICSE '04 Proceedings of the 26th International Conference on Software Engineering. 2004.
- From Representations to Computations: The Evolution of Web Architectures. Justin R. Erenkrantz, Michael Gorlick, Girish Suryanarayana and Richard N. Taylor. 2007.
- An Architectural Style for Solving Computationally Intensive Problems on Large Networks. Yuriy Brun and Nenad Medvidovic. Proceedings of the 2007 International Workshop on Software Engineering for Adaptive and Self-Managing Systems. 2007.

Bibliografia



- Tutoriais:
 - [Hadoop MapReduce Tutorial Series](#)
 - [AWS Architecture Center](#)
 - [Netflix Cloud Computing Tutorial](#)



Pós-Graduação em Computação Distribuída e Ubíqua

INF628 - Engenharia de Software para Sistemas Distribuídos

Sandro S. Andrade
sandroandrade@ifba.edu.br