

Ingegneria Dei Sistemi Distribuiti

- Materiale, link utili, avvisi
 - <http://www.dmi.unict.it/tramonta/sdm>
 - GitHub
 - Telegram

1

Prof. Tramontana - Ottobre 2024

Lezioni

- In Aula 36
 - Martedì 9:00–11:00
 - Giovedì 9:00–11:00
 - Laboratorio Venerdì 17:00-19:00
- Coprono tutto il programma del corso
- Partecipazione fortemente consigliata: si impara di più, e meglio, è possibile fare domande ed ottenere risposte!

2

Prof. Tramontana - Ottobre 2024

Programma

- Tecniche per sviluppare un sistema software distribuito o parallelo, di grandi dimensioni, che deve andare in produzione
- Buone norme di progettazione di tali sistemi
 - Design Pattern [Gamma]: Proxy
 - Design Pattern [Buschmann]: Remote Proxy, Forwarder-Receiver, Broker

3



Programma

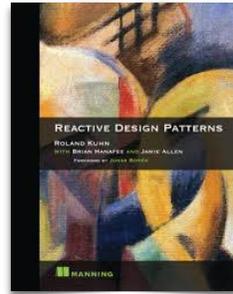
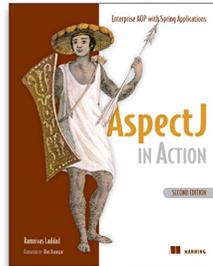
- Design Pattern [Fowler]: Remote Facade, Data Transfer Object, Session State, Serialized LOB, Gateway, Registry
- Design Pattern per la sicurezza [Schumacher]: Access Control (Reference Monitor, Role-Based Access Control), Authenticator

4



Programma

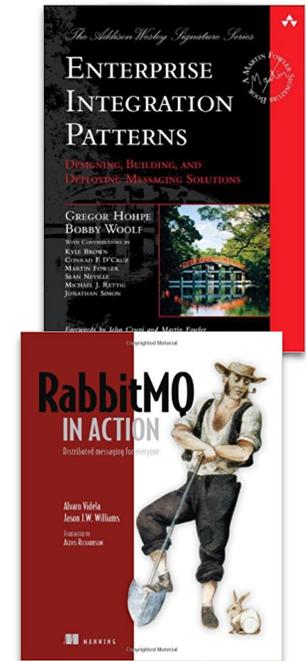
- Time-Out, Latenze e Design Pattern Circuit Breaker e Bulkhead
- Java CompletableFuture per programmazione asincrona e timeout
- Aspect-Oriented Programming (AOP)



Prof. Tramontana - Ottobre 2024

Programma

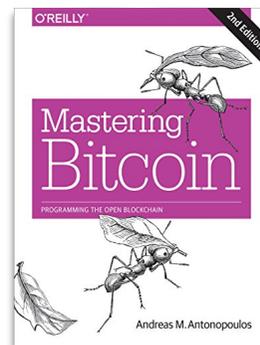
- Middleware orientato al Message-Passing, RabbitMQ
- Design pattern: event-driven consumer, competing consumers, publish-subscribe, message-filter



6

Programma

- Architettura distribuita di una blockchain per il consenso e per la registrazione di transazioni
- Piattaforma Ethereum per la programmazione degli smart contract



7

Prof. Tramontana - Ottobre 2024

Libri Consigliati

- Le slide sono di supporto. Gli appunti presi a lezione, insieme alle slide potrebbero bastare
- Buschmann, et al. Pattern-Oriented Software Architecture. A System of Patterns. Volume 1. Wiley, 1996
- Fowler. Patterns of Enterprise Application Architecture. Addison-Wesley, 2003
- Gamma, et al. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994
- Hohpe, et al. Enterprise Integration Patterns. Addison-Wesley, 2003
- Kuhn, et al. Reactive Design Patterns. Manning, 2017
- Laddad. AspectJ in Action. Manning, 2nd ed, 2010
- Schumacher, et al. Security Patterns. Wiley, 2006
- Urma, et al. Java 8 in Action: Lambdas, streams, and functional-style programming. Manning, 2015
- Videla, et al. RabbitMQ in Action. Manning, 2012

Prof. Tramontana - Ottobre 2024

Modalità Esami

- Progetti individuali o per coppie + Orali
- I Progetti saranno concordati, a partire da circa metà del corso
 - Si dovrà produrre
 - Relazione con: descrizione requisiti, diagrammi UML, dettagli di progettazione e implementazione
 - Codice applicazione e codice test
- Valutazione
 - Uso di appropriate soluzioni architetture (ovvero modularità e design pattern)
 - Chiarezza della documentazione