

# Fundamentos de Sistemas Distribuídos

## Trabalho Prático

2018/2019

### Informações gerais

- Cada grupo deve ser constituído por três elementos.
- Não deve ser apresentada uma interface para o utilizador, apenas uma interface para o programador de aplicações (API) e exemplos da sua utilização.
- Deve ser entregue o código fonte e ainda um relatório até 6 páginas em formato pdf.
- O trabalho deve ser entregue até 4 de janeiro de 2019 no *e-Learning*.

### Objetivo

Desenvolvimento de um sistema distribuído de armazenamento persistente de pares chave–valor. O trabalho deve ser elaborado utilizando Java e Atomix (apenas as classes já usadas nas aulas).

### Requisitos

1. O sistema deve usar vários servidores para armazenar de forma persistente chaves (Long) a que são associados valores (`byte[]`) oferecendo as seguintes operações aos clientes:
  - `CompletableFuture<Boolean> put (Map<Long, byte[]> values):` Escreve um conjunto de pares chave–valor, indicando se teve sucesso.
  - `CompletableFuture<Map<Long, byte[]> get (Collection<Long> keys):` Lê os valores associados a um conjunto de chaves.
2. O sistema proposto deve garantir que ou todos os pares chave–valor incluídos no pedido de escrita são modificados, ou então nenhum é modificado, mesmo que haja falha e reinício de alguns servidores. No entanto não se colocam quaisquer restrições ao que leitores concorrentes observam. Admite-se que a operação de leitura bloqueie ou devolva uma exceção em caso de falha e reinício dos servidores.

### Valorização

1. Adicionalmente garanta que no caso de duas escritas concorrentes nas mesmas duas ou mais chaves, mesmo que armazenadas em servidores distintos, os valores escritos em último lugar nas diferentes chaves provêm da mesma operação.
2. A resolução proposta deve ser modular, maximizando o código que é independente desta aplicação em concreto, e adequado à utilização em grande escala, sem gargalos ou limitações, devidamente justificado no relatório.