

Disciplina Programação Modular	Departamento Engenharia de Software	Turno Manhã/Noite	Período 2º
Professor Hugo de Paula (hugo@pucminas.br)			

## Trabalho Prático sobre Princípios SOLID e Padrões de Projeto em dupla

### 1 Objetivos do trabalho

O objetivo deste trabalho é que os conceitos aprendidos ao longo da disciplina de Programação Modular sejam aplicados em conjunto em um programa coeso e extensível, que utilize padrões de projeto.

O sistema deve ser desenvolvido em dupla.

O sistema a ser desenvolvido está detalhado na seção a seguir.

### 2 O problema

Neste trabalho deverá ser desenvolvido um sistema para uma bolsa de valores qualquer, como a Bovespa, utilizando padrões de projeto. Quem quiser conhecer um pouco mais sobre o funcionamento da bolsa de valores, visite o site da BOVESPA em [http://www.b3.com.br/pt\\_br/market-data-e-indices/servicos-de-dados/market-data/cotacoes/](http://www.b3.com.br/pt_br/market-data-e-indices/servicos-de-dados/market-data/cotacoes/).

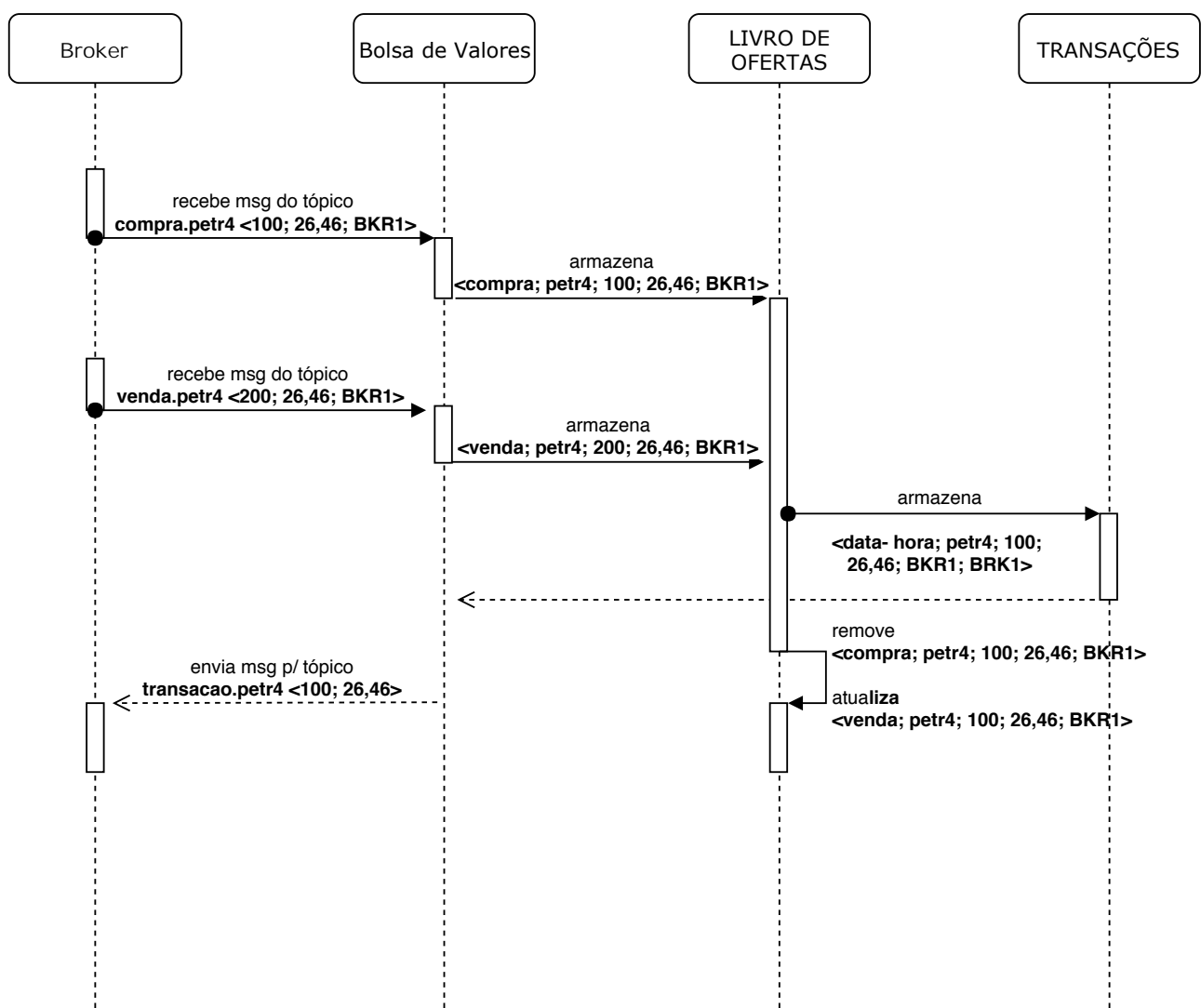
A corretora (ou *broker*) pode enviar as seguintes operações à bolsa de valores:

OPER.	FORMATO	DESCRIÇÃO
compra	<quant: int, val: real, corretora : char[4]>	Envia ao servidor da bolsa de valores uma ordem de compra indicando que a <i>corretora</i> deseja comprar <i>quant</i> lotes de ação pelo preço de <i>val</i> reais.
venda	<quant: int, val: real, corretora : char[4]>	Envia ao servidor da bolsa de valores uma ordem de venda indicando que a <i>corretora</i> deseja vender <i>quant</i> lotes de ação pelo preço de <i>val</i> reais.
info	<data-hora: char[16] "dd/mm/aaaa hh:mm">	Envia ao servidor da bolsa de valores uma solicitação de informações de uma ação na data especificada.

As regras para comunicação entre os módulos são:

- *Brokers* podem enviar ORDENS DE COMPRA, ORDENS DE VENDA e INFO para a bolsa de valores.
- *Brokers* podem assinar tópicos relativos às ações que desejam acompanhar.
- Sempre que a bolsa de valores recebe ordem de compra ou de venda, ela deve encaminhar essa ordem a todos os *brokers* interessados naquela ação específica.
- Sempre que o valor de uma ORDEM DE COMPRA for maior ou igual ao valor de uma ORDEM DE VENDA para uma mesma ação, a bolsa de valores deve gerar uma *transacao*  $\langle quant: int, val: real \rangle$ , e atualizar/remover as ordens da fila.

O diagrama a seguir exibe um cenário de interação entre a bolsa e duas corretoras:



O sistema deve ser carregado com a lista de ativos da Bovespa. A tabela a seguir ilustra alguns exemplos de ativos da Bovespa.

Nome de Pregão	Código	Atividade Principal
AMBEV S/A ON	ABEV3	Fabricação e Distribuição de Cervejas. Refrigerantes e Bebidas Não Carbonatadas e Não Alcoólicas.
PETROBRAS PN	PETR4	Petróleo. Gás e Energia
VALE PNA	VALE5	Mineração
ITAUUNIBANCO PN	ITUB4	A Sociedade Tem por Objeto A Atividade Bancária.
BRADESCO PN	BBDC4	Prática de Operações Bancárias em Geral. Inclusive Câmbio
BRASIL ON	BBAS3	Banco Múltiplo.
CIELO ON	CIEL3	Empresa Prestadora de Serviços de Adquirência e Meios de Pagamento.
PETROBRAS ON	PETR3	Petróleo. Gás e Energia.
HYPERMARCAS ON	HYPE3	Produção e Venda de Bens de Consumo e Medicamentos.
VALE ON	VALE3	Mineração
BBSEGURIDADE ON	BBSE3	Participação no Capital Social de Outras Sociedades. que Tenham por Atividade Operações de Seguros. Resseguros. Previdências Complementar ou Capitalização.
CETIP ON	CTIP3	Sociedade Administradora de Mercados de Balcão Organizados.
GERDAU PN	GGBR4	Participação e Administração.
FIBRIA ON	FIBR3	
RAIADROGASIL ON	RADL3	Comércio de Produtos Farmacêuticos. Perfumarias e Afins.

O trabalho consiste em desenvolver um pequeno aplicativo que simule a interação do *Broker* com a *Bolsa de valores*, utilizando padrões de projeto. Os requisitos do trabalho são:

1. As informações das ações devem ser centralizadas em apenas uma única *bolsa de valores*.
2. Vários *brokers* podem observar múltiplas ações na *bolsa*.
3. A *bolsa* deve notificar os *brokers* das atualizações no livro de ofertas e as operações realizadas em uma ação.
4. Os *brokers* devem ser capazes de enviar ofertas para a *bolsa*.

### 3 Requisitos não funcionais

1. Esse trabalho usará pelo menos três dos padrões de projeto: **Singleton**, **Decorator**, **Factory** e **Observer**.
2. Será necessário utilizar Threads em Java para simular múltiplos *brokers* acessando a *bolsa*.

Será avaliada a conformidade com os requisitos, bem como a qualidade do código, utilização dos conceitos básicos e avançados de programação modular, tais como Collections, Tratamento de exceções, etc. Não é necessário implementar interface gráfica com o usuário.

O sistema deverá ser projetado utilizando a notação UML. O sistema deverá ser testado com testes unitários baseados na JUnit.

## 4 Resultados Esperados

Deverá ser entregue em um arquivo comprimido em formato .zip:

1. Todo o código, comentado.
2. Um arquivo README.TXT contendo uma explicação sucinta do código e instruções para compilação e execução do mesmo.
3. Uma documentação em PDF descrevendo a estrutura do sistema: classes, operações, etc. É obrigatória a apresentação de um diagrama de classes no formato UML para ilustrar o sistema e um diagrama de componentes para ilustrar a arquitetura.

O nome do arquivo .zip deve ser a concatenação dos nomes dos integrantes do grupo.