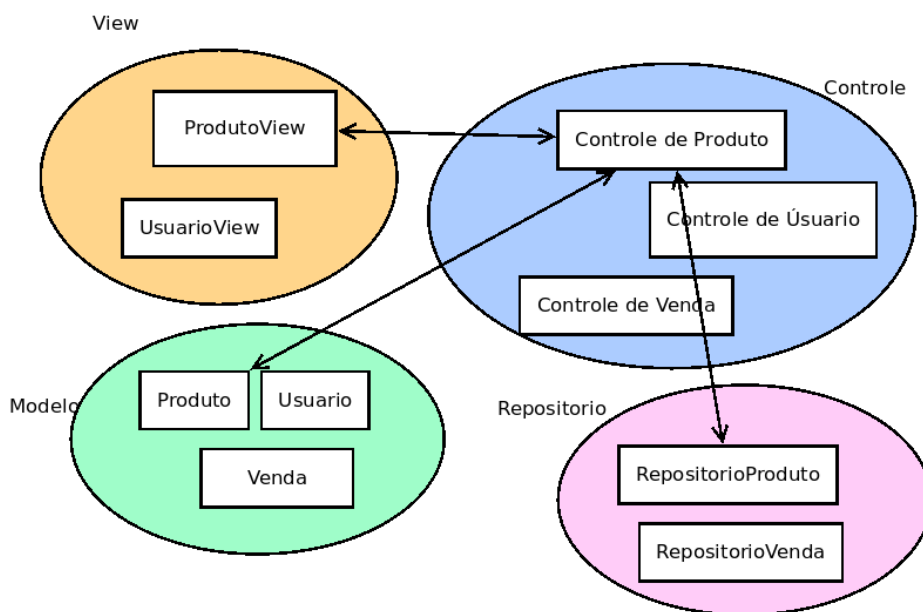
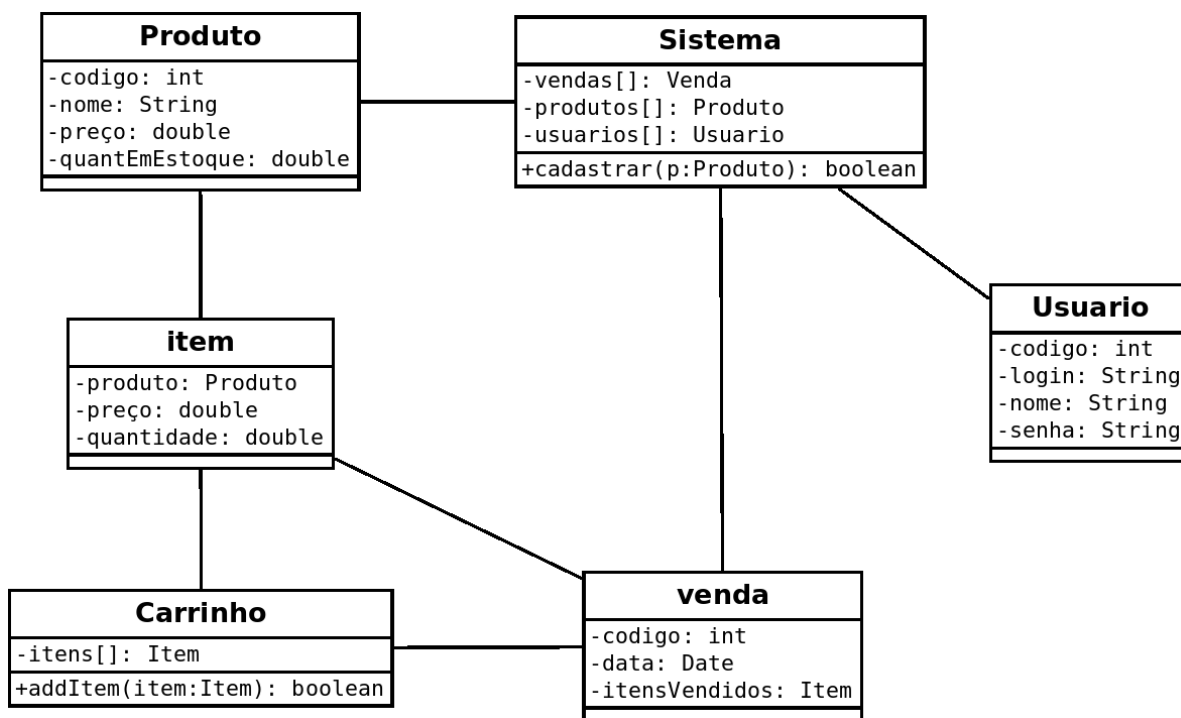


A partir do trabalho desenvolvido anteriormente, modifique-o aplicando a arquitetura de software MVC-like vista em aula.



Também será necessário acrescentar um sistema de Controle de login e CRUD de usuários. Segue uma sugestão de modelagem.



**Requisitos:**

- Método `init` em `Sistema` que inicializa a aplicação com 5 produtos e 2 vendas;
- Não deve ser possível inserir mais de um produto com o mesmo nome;
- Não deve ser possível excluir produto que tenha vendas realizadas. Neste caso, o produto é marcado como excluído (não é efetivamente excluído), mas e não aparece na listagem de produtos;
- Ao inserir produtos no carrinho, produtos de mesmo código devem ser aglutinados como um único item;
- Não deve ser possível concretizar venda que possa tornar a quantidade de um produto negativa;
- Aplique os conceitos de encapsulamento e acoplamento;
- As classes devem ter construtores e métodos fábrica para permitir a criação de objetos com estado inicial válido.
- Não deve ser possível criar mais de um objeto `Sistema` (singleton);
- Listagens no formato de tabela: cada objeto é uma linha desta tabela. A largura da coluna deve ser configurável. Por Exemplo:

<i>CÓDIGO</i>	<i>NOME</i>	<i>MARCA</i>	<i>PREÇO</i>	<i>QUANTIDADE</i>
<i>1</i>	<i>Arroz</i>	<i>Prato fino</i>	<i>15,00</i>	<i>1000</i>
<i>2</i>	<i>Macarrão</i>	<i>Mara</i>	<i>3,50</i>	<i>600</i>

**OBSERVAÇÕES:**

- O trabalho poderá ser feito em dupla;
- O trabalho deve ser desenvolvido na linguagem Java com uso de boas práticas de programação orientada a objetos;
- Lembre-se que um método representa um comportamento de objetos de uma classe e tem apenas uma única responsabilidade;
- Métodos *static* são a exceção e não a regra. Desconfie se seu trabalho tiver muitos métodos estáticos.

BOM TRABALHO!