

## Documentação do código fonte

O sistema é feito em arquitetura MVC(model-view-controller) utilizando uma temática simples e fácil.

### O que é MVC

MVC é um tipo de arquitetura que separa as responsabilidades de cada tipo de regra de negócios nos seus respectivos tipos.

model é responsável pela camada de manipulação de dados onde são feitos as funções básicas de um programa tipo CRUD\* e toda a manipulação de dados

View é a camada de visualização, em que geralmente ficam disponíveis as “telas” feitas para interação com o usuário final

Controller é a camada da acesso a todos os dados que trafegam entre a camada view e a camada model, ele é responsável pela conexões com o banco de dados (se houve) e configurações do sistema

*\*CRUD (Create-read-update-delete) são as ações básica que quase todos os sistemas de computadores fazer. São eles inserir informações, retornar ou ler dados, atualizar e deletar dados através de operações no banco de dados ou em memória.*

## Classes

As classes contidas no sistema estão dividias em pastas ou pacotes (packages).

### Controller

-clienteDAO

-conectar

### Model

-cliente

-ItemPedido

-Pedido

-Produto

-Status

*View*

*-Main*

*-Pedido*

*ClienteDAO*

Nessa classe são realizados os métodos básicos do CRUD como inserir, excluir, alterar e listar todos

*Conectar*

Nessa classe estão todas as conexões com o banco de dados bem como os parâmetros para a configurações.

*Cliente, itemPedido, Produto*

Nessa classe estão os construtores, getters e setters responsáveis por manipular os dados relativos ao cliente, Itens dos pedidos e produto

*Status*

Essa classe é do tipo enumerável fazendo assim que o usuário escolha somente 3 tipos de opções. Ativo, processado e cancelado

*Main*

A classe principal do sistema onde todos os objetos são instanciados e onde há também a janela principal do sistema

*Pedido*

A última classe do sistema responsável por finalizar um pedido do cliente com todos os parâmetros iniciados pelo sistema