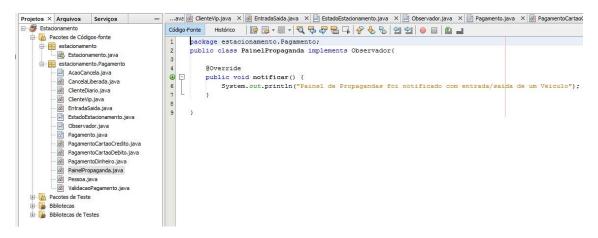
## Trabalho Design Patterns na Prática

## **Estacionamento**

Estacionamento trata-se de um sistema que auxilia o gerenciamento de uma empresa que visa tratar diferentes públicos, (rotativo, e os clientes vips). Foi usado para o desenvolvimento deste projeto 4 patterns sendo estes:

**Observer:** Foi implantando este pattern no projeto do estacionamento para que ele faça a contagem de entrada/saída de veículos do estacionamento, alertando em um painel de propagandas o fluxo de veículos. Este padrão notifica o painel assim que tem alguma alteração na cancela de entrada ou saída.



**Template:** O template por trabalhar com uma forma de definir os passos a serem seguidos foi implantado na parte da saída do cliente, fazendo com que ele não consiga sair sem que seja calculado o tempo estacionado, multiplicando assim o valor a ser pago pelo tempo consumido.

```
Projetos × Arquivos Serviços — 🔯 Estacionamento, java × 🖾 AcaoCancela. java × 🗹 Cancelal. java × 🚳 Cliente Diario, java × 🚳 Cliente Diario, java ×
Estacionamento
                                          Código-Fonte Histórico 🔯 👼 🕶 🔻 💆 🗸 🖓 🖶 📮 🗘 🔗 😓 🖄 💇 💇 🥚 🔲 🏙 🚅
   Pacotes de Códigos-fonte
                                                package estacionamento.Pagamento;
          Estacionamento.java
     estadoriamento, java
estadoriamento, Pagamento

AcaoCancela, java

CancelaLiberada, java

ClienteDiario, java

ClienteVip, java
                                          4 📮
                                                   public void recepcao() {
                                                      calcularTempo();
                                                         calcularValor();
                                                         validacao();
          EstadoEstacionamento.java
                                          10 public void calcularTempo() {
          Observador.java
                                          11 12
                                                        System.out.println("Calculando tempo que ficou estacionado..");
          Pagamento.java
           PagamentoCartaoCredito.java
                                          13 public void calcularValor() {
          PagamentoCartaoDebito.java
          PagamentoDinheiro.java
                                                       System.out.println("Calculando valor a ser pago..");
                                          14
15
          PainelPropaganda.java
                                                    protected abstract void formaPagamento();
                                           1
                                          17
          ValidacaoPagamento.java
                                          18 📮
                                                    public void validacao() {
  Pacotes de Teste
                                          19
                                                         AcaoCancela a = new ValidacaoPagamento();
   Bibliotecas
                                          20
                                                         a.comportamento();
   ⊕ 🔓 Bibliotecas de Testes
                                          21
```

**Proxy:** O proxy por sua vez faz a validação do meio de pagamento do cliente, dependendo de sua forma de pagamento a cancela muda seu status para aberta, ou permanece fechada e informa ao cliente que ele deve realizar outra forma de pagamento.

```
...ava 🗟 EstadoEstacionamento.java 🗴 🗟 Observador.java 🗴 🔯 Pagamento.java 🗴 🔞 PagamentoCartaoCredito.java 🗴 🚳 Pr
Projetos × Arquivos Serviços
                                         Código-Fonte Histórico 🔯 👨 - 💹 - 🍳 🐶 🖶 📮 🔗 😓 🔁 🖆 🎱 🔘 🕮 🚅

☐ Ø Estacionamento

  🖟 🚹 Pacotes de Códigos-fonte
                                               package estacionamento.Pagamento;
    estacionamento
         Estacionamento.java
                                               public class ValidacaoPagamento implements AcaoCancela {
     estacionamento.Pagamento
         AcaoCancela,iava
          CancelaLiberada.java
                                          (I)
                                                    public void comportamento() {
          dienteDiario.java
                                                        System.out.println("Verificando forma de Pagamento");
          ClienteVip.java
          EntradaSaida.java
                                                             CancelaLiberada livre = new CancelaLiberada();
          EstadoEstacionamento.java
                                         10
                                                             livre.comportamento();
          B Observador.java
          Pagamento.java
                                         12
                                                            System.out.println("Catraca fechada, sua forma de pagamento nao e "
          PagamentoCartaoCredito.java
                                         13
                                                                     + "valida, tente outra forma de pagamento");
          PagamentoCartaoDebito.java
          PagamentoDinheiro, java
                                         15
          PainelPropaganda.java
                                         16
          Pessoa.java

    ValidacaoPagamento.java

  ⊕ 🌇 Pacotes de Teste
    Bibliotecas
  Bibliotecas de Testes
```

**State:** O state viabiliza o controle de que se o estacionamento está vazio ou lotado. E necessário este tipo de tratamento pois se tem dois tipos de clientes, necessitando assim estados diferentes de execução de ações para clientes de diferentes classificações. Se o cliente for vip, o estacionamento estando lotado, ele e direcionado a uma área onde poderá estacionar seu veículo, caso o cliente seja rotativo (diário) e o estacionamento estiver lotado, ele e abortado com uma mensagem de desculpas e logo informa que está lotado no momento.

```
Projetos X Arquivos Serviços — 🚳 Estacionamento, java X 🖾 AcaoCancela. java X 🚳 Cancelat. iberada. java X 🔯 Cliente Diario. java X
Estacionamento
                                       Código-Fonte
                                                  Histórico
                                                           Pacotes de Códigos-fonte
                                             package estacionamento.Pagamento;
    estacionamento
                                             public class ClienteDiario implements EstadoEstacionamento{
         Estacionamento, java
                                        3
    estacionamento.Pagamento
                                                  @Override
         AcaoCancela.java
CancelaLiberada.java
                                        ③ □
                                                  public void vazio() {
                                                      System.out.println("Estacionamento vazio!");
                                           L
         & ClienteDiario.java
         ClienteVip.java
         EntradaSaida.java
                                                  @Override
         EstadoEstacionamento.java
                                        ① 🗇
                                                  public void lotado() {
         Observador.java
                                                      System.out.println("Desculpe o transtorno, estacionamento Lotado");
                                        11
         Pagamento.java
         PagamentoCartaoCredito.iava
                                        13
         PagamentoCartaoDebito.java
                                        14
                                             }
         PagamentoDinheiro.java
                                        15
         PainelPropaganda.java
         Pessoa.java
         WalidacaoPagamento.java
  Pacotes de Teste
  Bibliotecas
```

Cliente Diário.

```
Projetos × Arquivos Serviços — 🖟 Estadonamento, java × 🖂 AcaoCancela, java × 🗟 Cancelal, berada, java × 🗟 Cliente Diario, java × 🔞 Cliente Diario, java ×
  Estacionamento

Estacionamento

Estacionamento

Estacionamento

Estacionamento.java
                                                                  Código-Fonte Histórico 🔯 👼 - 💹 - 🍳 🕏 ኞ 🖶 📮 🔐 🔗 😓 🖆 🖭 🎱 📵 🛍 🚅
                                                                            package estacionamento. Fagamento;
public class ClienteVip implements EstadoEstacionamento{

Statadonamento, java
estadonamento, java
estadonamento, java
AcaoCancela, java
Cancelalberada, java
ClienteDiario, java
ClienteVipi, java
EstadoEstadonamento, java
Deservador, java
Deservador, java
                                                                                   public void vazio() {
                                                                                         System.out.println("Estacionamento vazio!");
                                                                                   @Override
                                                                  9
11
12
13
14 }
                                                                                  public void lotado() {
                                                                                        System.out.println("Estacionamento lotado, dirija-se para area dos clientes Vips");
                 Pagamento.java
PagamentoCartaoCredito.java
                 PagamentoCartaoDebito.java

PagamentoCartaoDebito.java

PagamentoDinheiro.java

PainelPropaganda.java

Pessoa.java

■ ValidacaoPagamento.java
      Pacotes de Teste
Bibliotecas
Bibliotecas de Testes
```

Cliente Vip.