**ADAPTER**

→ Conceito:

O padrão de projeto estrutural adapter consiste na criação de uma classe que será usada para adaptar mudanças realizada no sistema, permitindo que um objeto seja substituído por outro que realiza a mesma tarefa mas com outra interface. Com o padrão adapter é possível alterar pontos específicos do código do sistema que sofreram modificações.

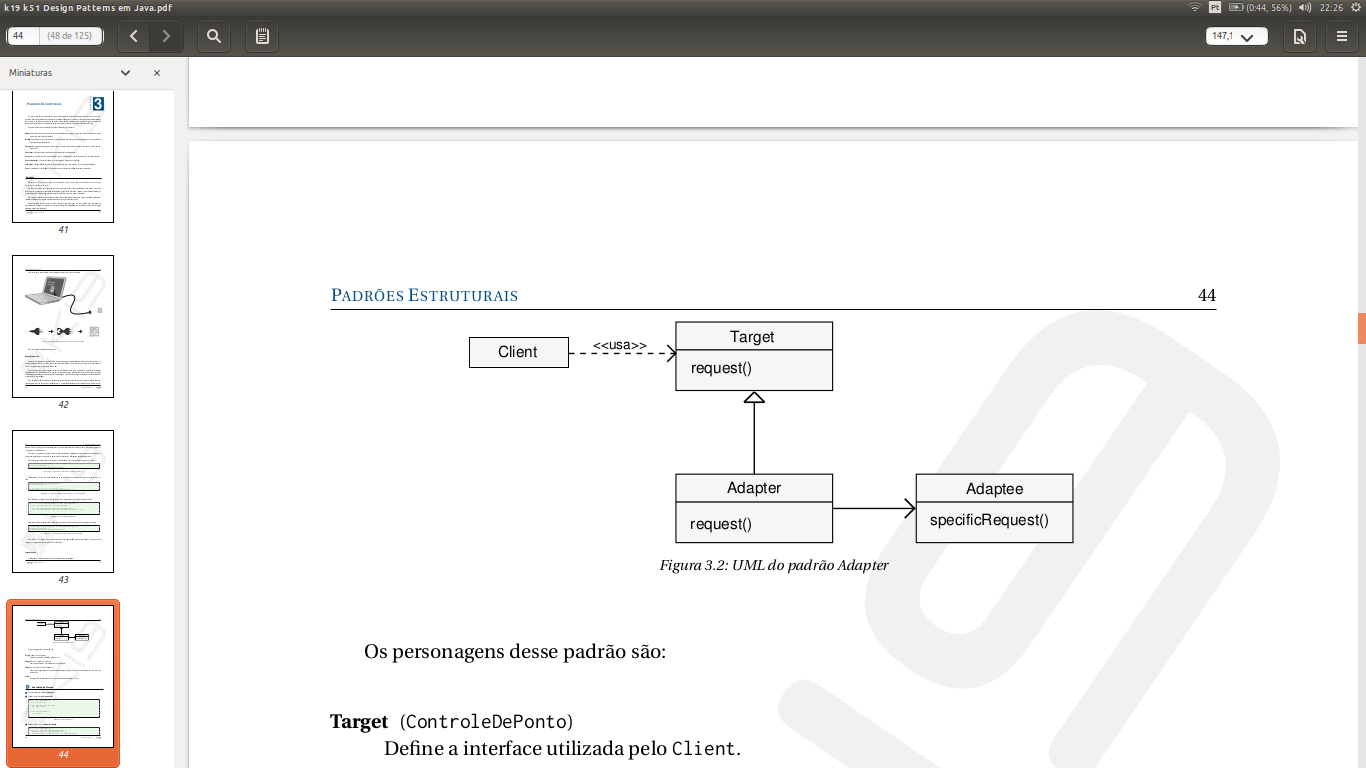
O adapter é usado principal quando é necessário implementar uma nova biblioteca em um sistema de software já existente e elas se diferem, como não é possível alterá-las então é criada uma classe que realiza essa adaptação responsável por adaptar a interface da nova biblioteca no formato que o sistema operada.

→ Aplicação:

A principal aplicação do padrão adapter ocorre quando se quer compatibilizar o sistema existente a outros frameworks e APIs (interface de programação de aplicativos). Quando temos a mudança da interface de um fornecedor para uma interface que o cliente deseja, sua implementação se torna complexa de acordo com a complexidade da interface do fornecedor. Suas principais vantagens são: composição de objetos,vinculação do cliente a uma interface e não a uma implementação*.* O padrão adapter é usado quando:

* se deseja usar uma classe existente mas sua interface não corresponde a necessidade real;
* é criado uma classe reutilizável que coopera com classes não relacionada ou não prevista ;
* (para adaptadores de objetos) é preciso usar várias subclasses existentes porém não é viável adaptar interface, criando subclasse para cada uma, então um adaptador de objeto pode adaptar a interface da sua classe-mãe.

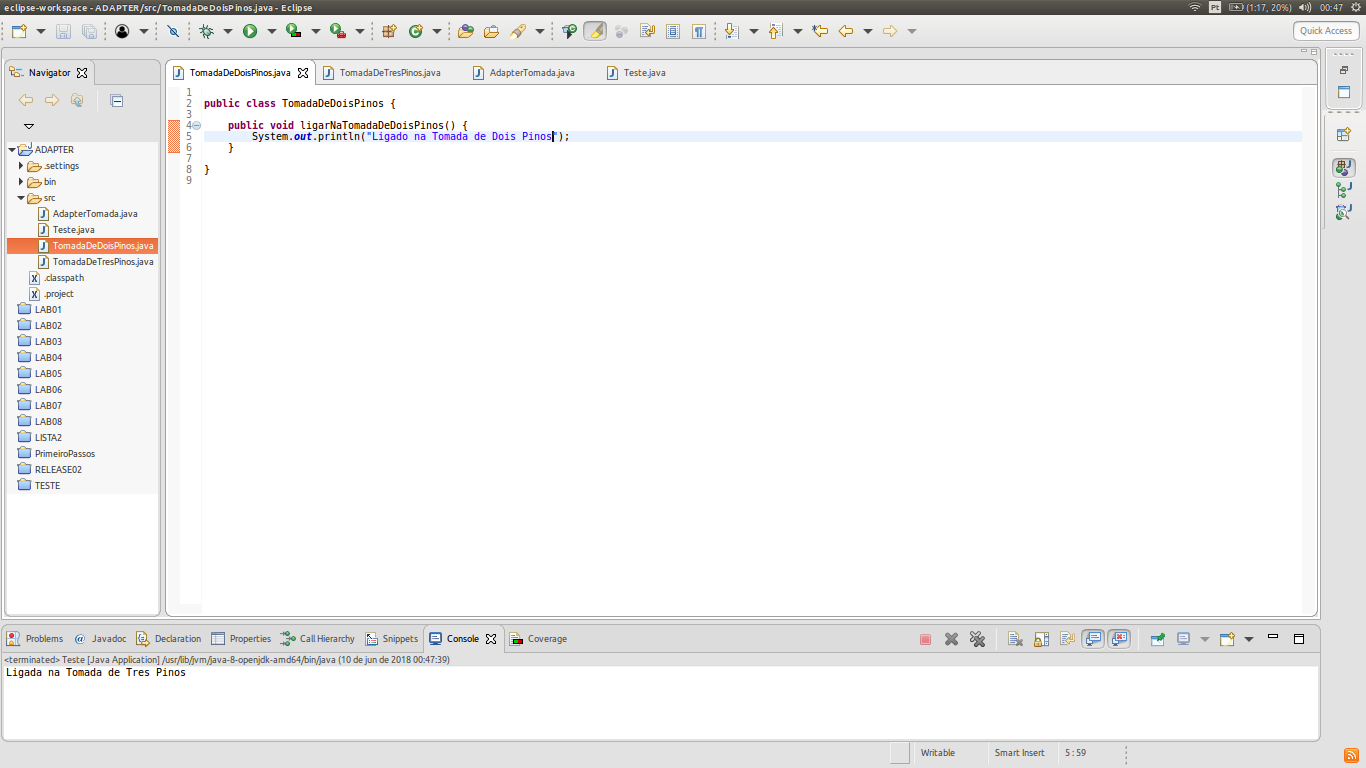
→Diagrama UML Padrão Adapter:



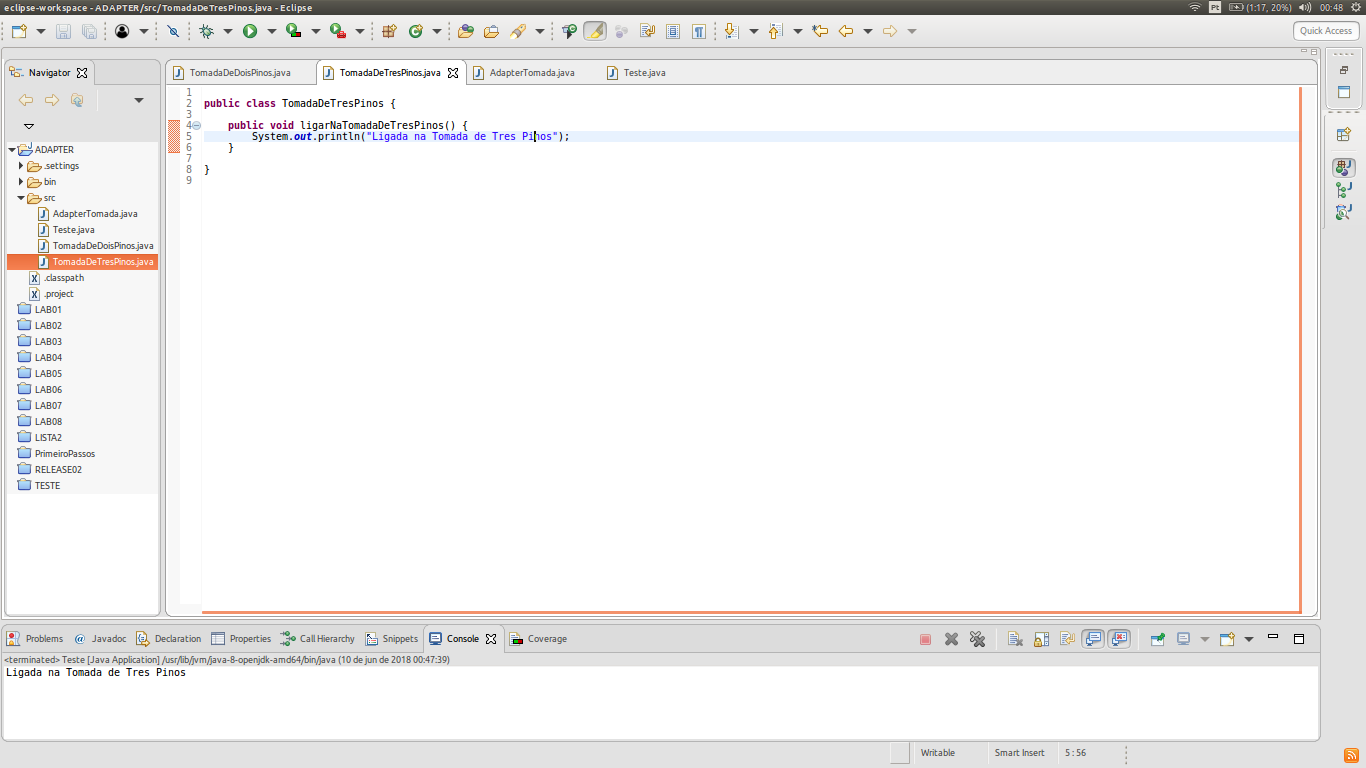
→ Exemplo:

A modelagem que será exemplificada teremos uma uma classe TomadaDeDoisPinos que precisa ser adaptada para uma classe TomadaDeTresPinos que possui método diferente da classe já existente. Então é criada a classe Adapter para ter acesso ao método desejado (TomadaDeTresPinos)

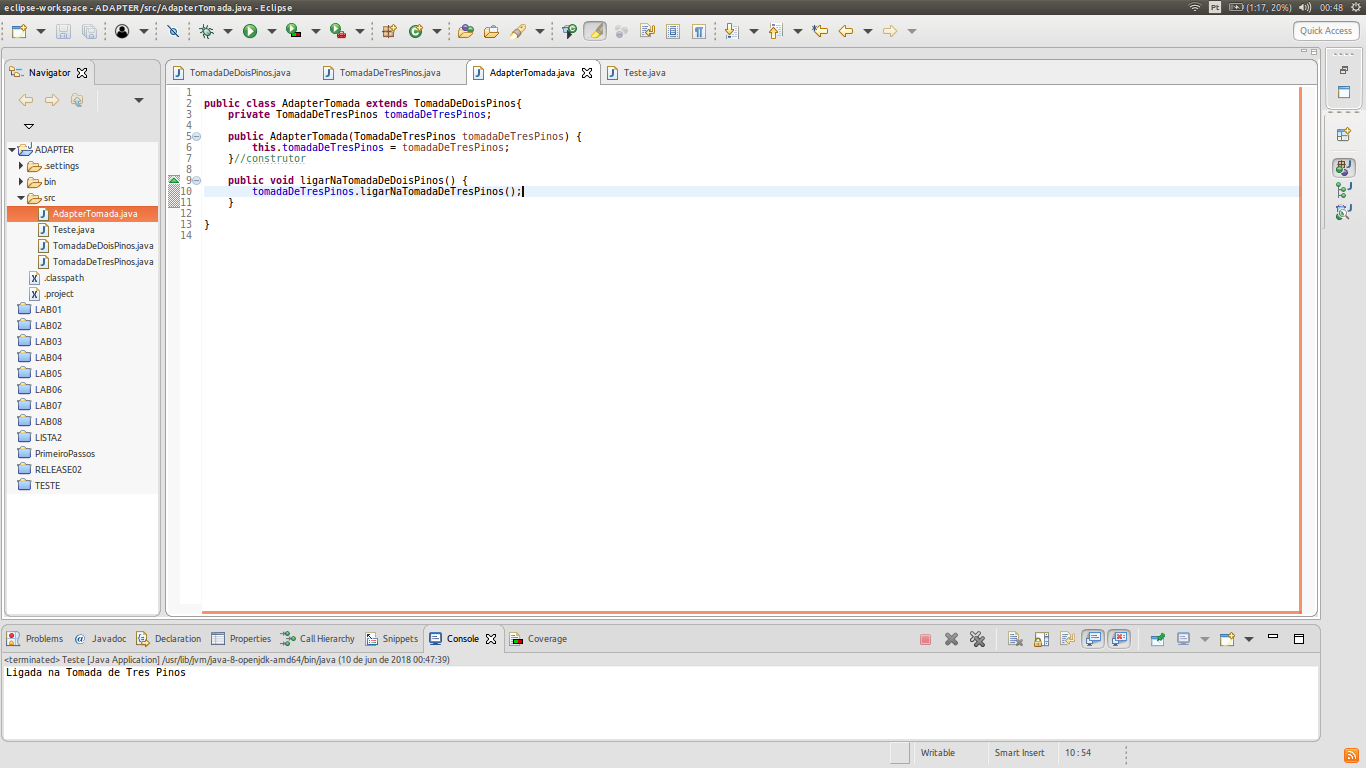
**Targert**: Define a interface utilizada pelo Cliente.



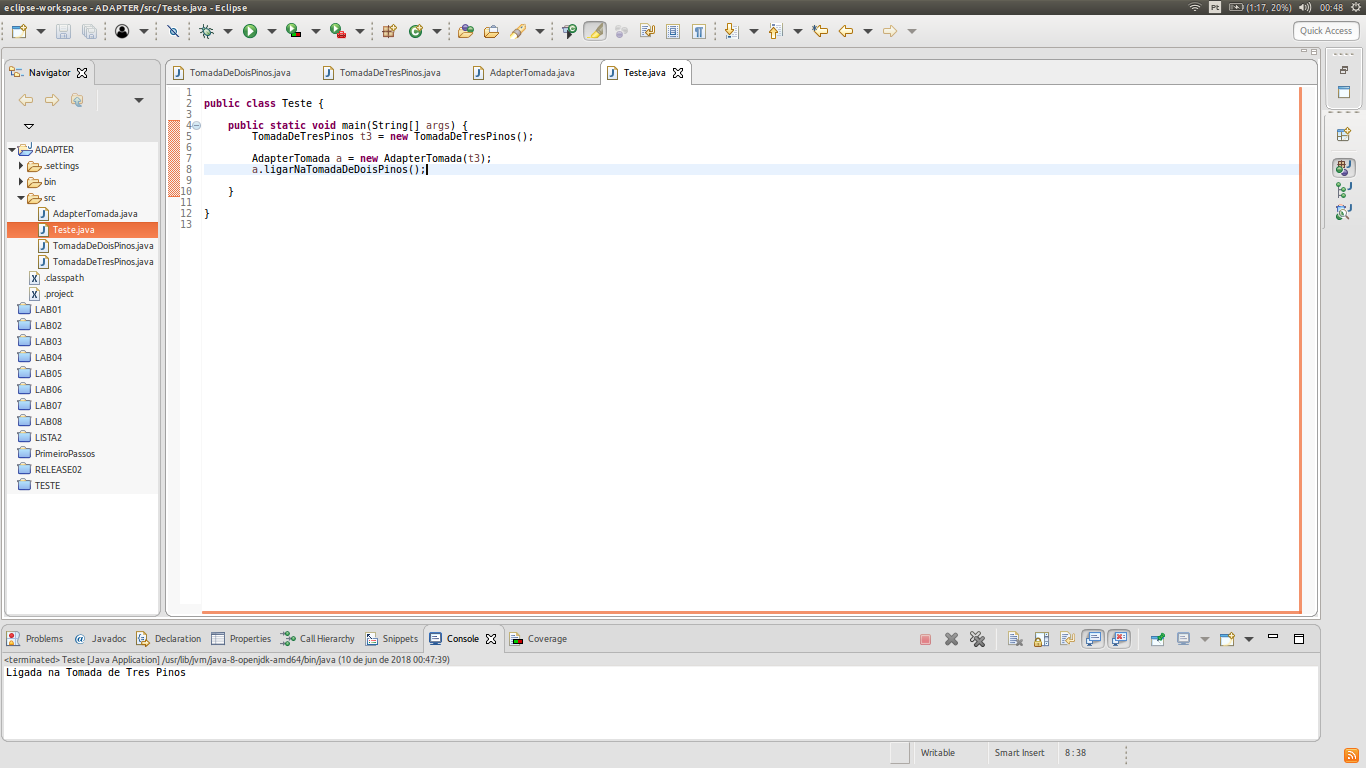
**Adaptee**: Classe que define o novo objeto a ser utilizado.



**Adapter**: Classe que implementa a interface definida pelo Target e adapta as chamadas do Cliente para o Adaptee.



**Cliente**: Interage com os objetos através da interface definida por Target.



→ Processo:

Os clientes chamam operações em uma instância de Adapter e este chama as operações de Adaptee que executa a solicitação.

→ Adapter de Classe:

Modifica Adaptee a Target através de uma classe Adapter oncreta, como consequência não é possível adaptar uma classe e suas subclasses; é possível que em Adapter seja substituído algum comportamento de Adaptee, devido a herança entre essas classes;

→ Adapter de Objeto:

Um único Adapter pode influenciar Adaptee e suas subclasses e também pode acrescentar funcionalidade a todos os Adaptee em uma única vez.