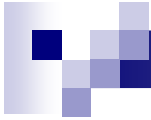


Laboratório de ENGSOE – Estudo de Caso



Laboratório ENGSOE

- Aula de Hoje:
 - Desenvolver um sistema UML inteiro:
 - Aplicação Banco Online.

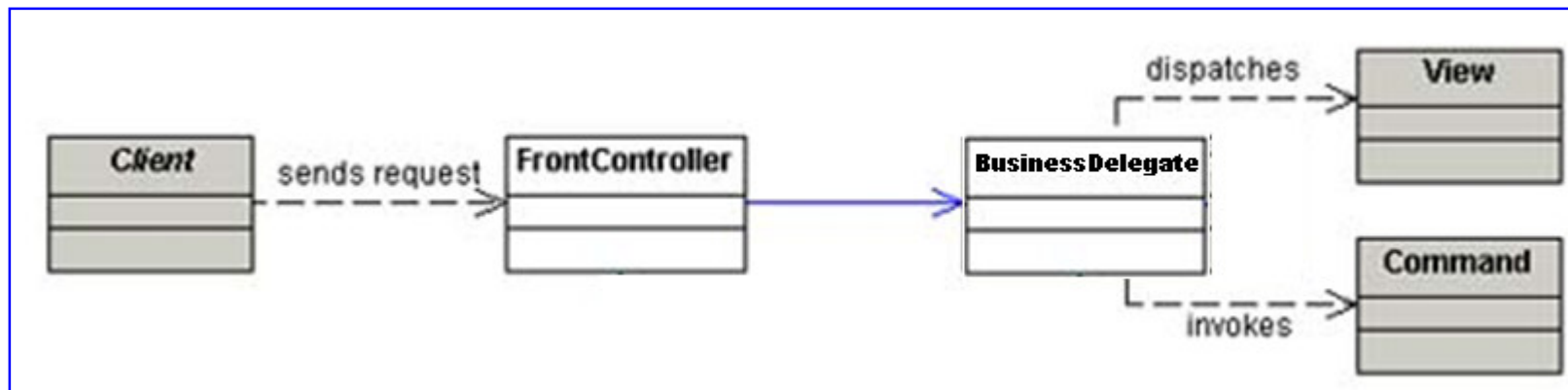


Laboratório ENGSO

- Nosso “Estudo de Caso!”
 - RSA V7 – O que será feito para o projeto?
 - 1) Criando um Projeto UML:
 - 1) Fase de Design (Desenvolver o Projeto do Sistema):
 - 1) Design do projeto
 - 2) Diagrama de classes de design
 - 3) Aplicação de Design Patterns na arquitetura da aplicação
 - 4) Diagrama de componentes
 - 5) Diagrama Físico

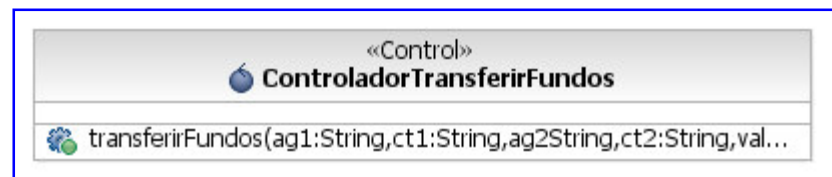
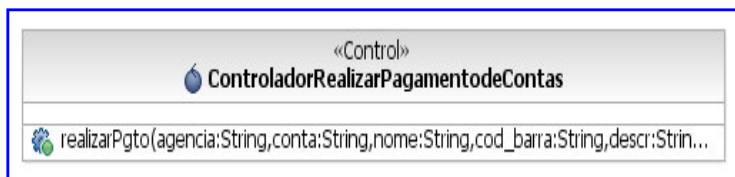
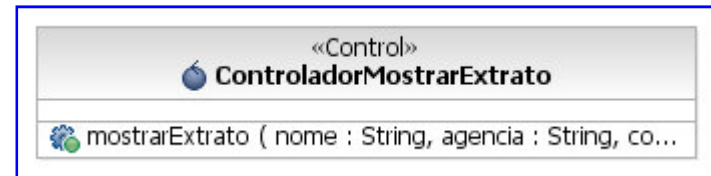
Laboratório ENGSOE

- Estruturando a aplicação.
 - Aplicaremos mais padrões para a aplicação:
 - Front-Controller: É o ponto de contato inicial para tratar todas as solicitações no sistema. Permite a um auxiliar a concluir a autenticação e autorização de um usuário ou iniciar a recuperação de contatos
 - Business Delegate: Separa a camada de apresentação da camada de negócio.
 - View: Uma página WEB (Ex: JSP de resposta)
 - Command: O componente de negócio.



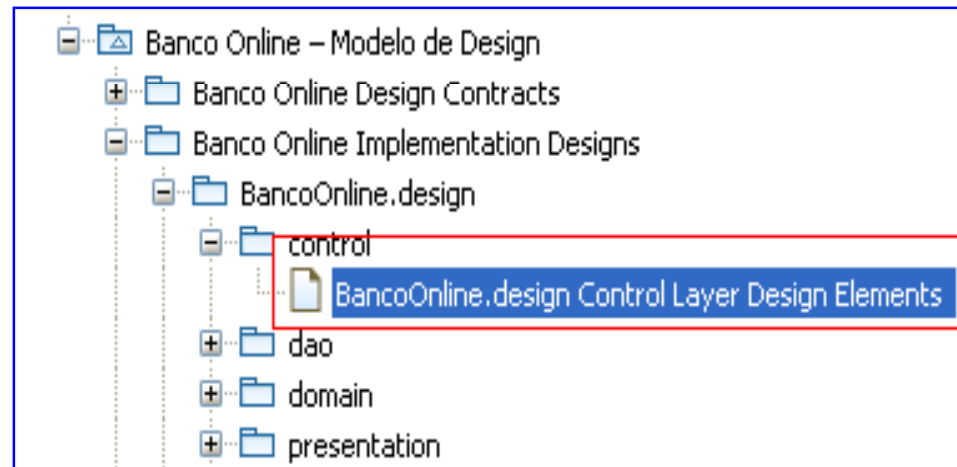
Laboratório ENGSOE

- Estruturando a aplicação.
- Será criado uma nova classe “FrontController” que irá ficar na frente dos diversos controladores abaixo.
- Refatoração: Cada controlador possui características de Controladores e de Logica de Negócio. Vamos separar em:
 - Business Delegate (Command Pattern)
 - EJB3



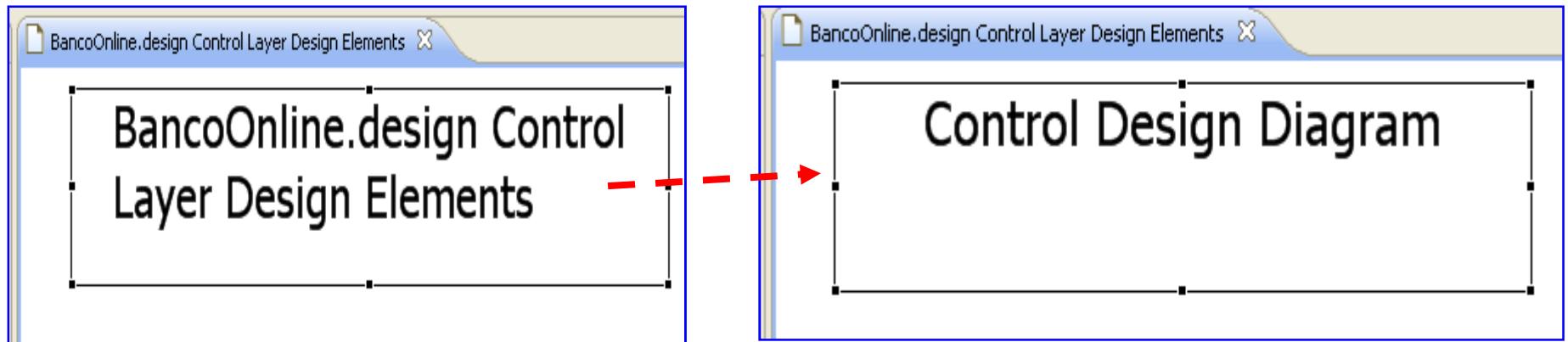
Laboratório ENGSOFT

- Estruturando a aplicação.
- Utilização do pacote “control”.
 - Expandir “Banco Online – Modelo de Design”, em “Banco Online Implementation Designs”, abrir o diagrama “BancoOnline.design Control Layer Design Elements” para “Diagrama de dominio do design”.



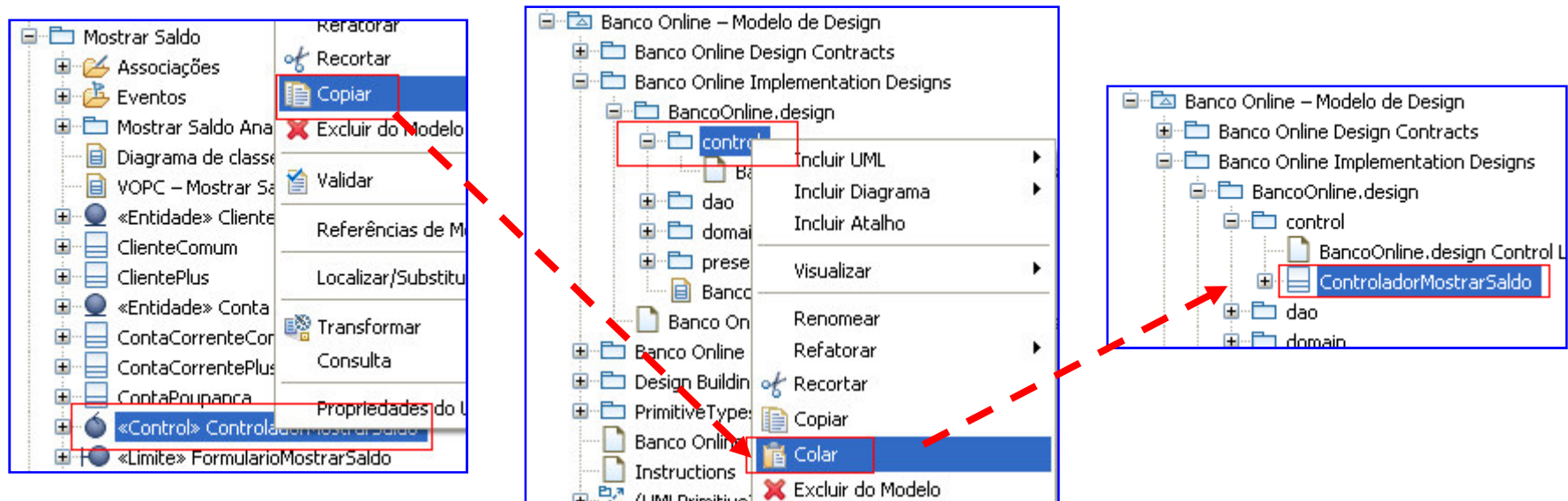
Laboratório ENGSOE

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
 - Abra o diagrama “BancoOnline.design Control Layer Design Elements”.
 - Renomear o nome do diagrama “BancoOnline.design Control Layer Design Elements” para “Control Design Diagram”.Dê um “F2” em cima do “label”.



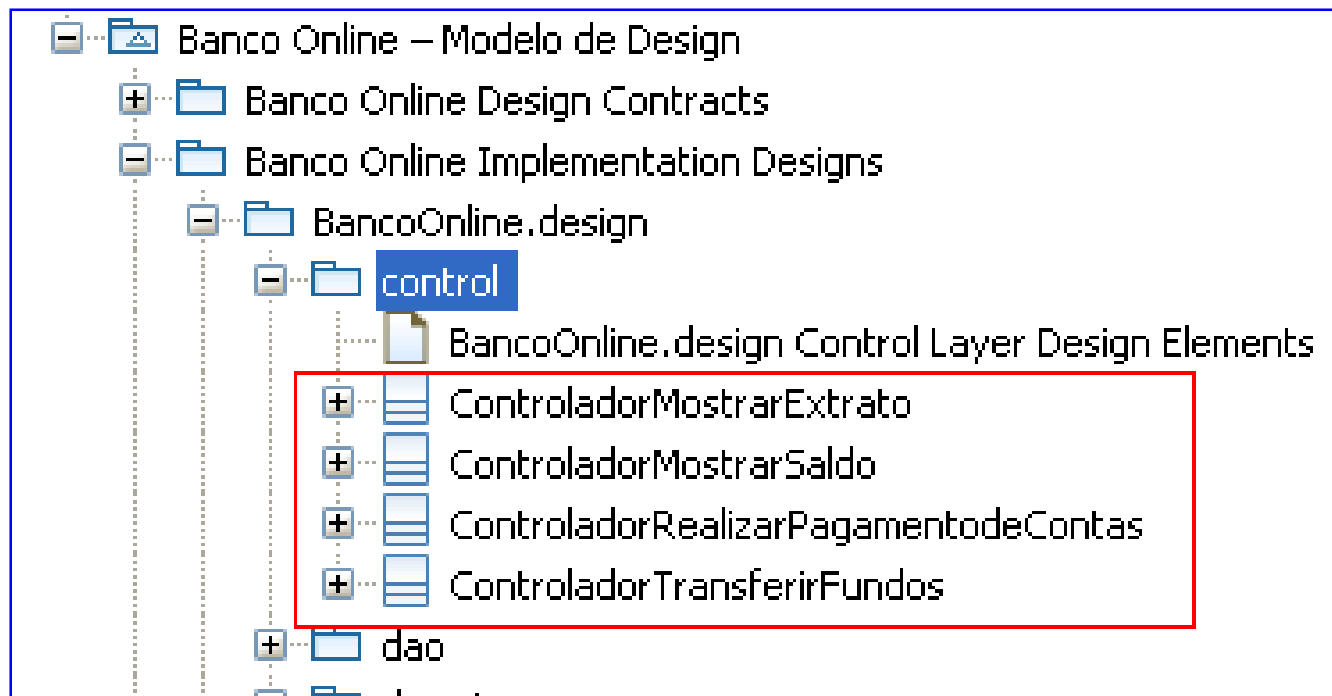
Laboratório ENGSOFT

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
- Abra o modelo “Banco Online – Modelo de Análise”->”Mostrar Saldo”. Clique com o botão direito em “ControladorMostrarSaldo” e escolha “Copiar”.
- Abra o modelo “Banco Online – Modelo de Design”->” Banco Online Implementation Designs”->” BancoOnline.design”->”control”. Clique com o botão direito em “control” e escolha “Colar”.



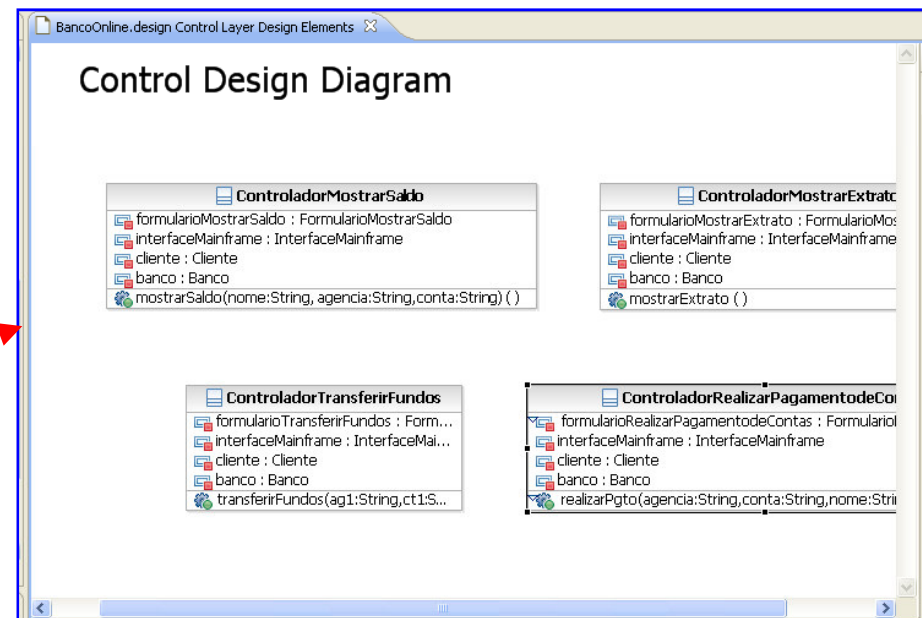
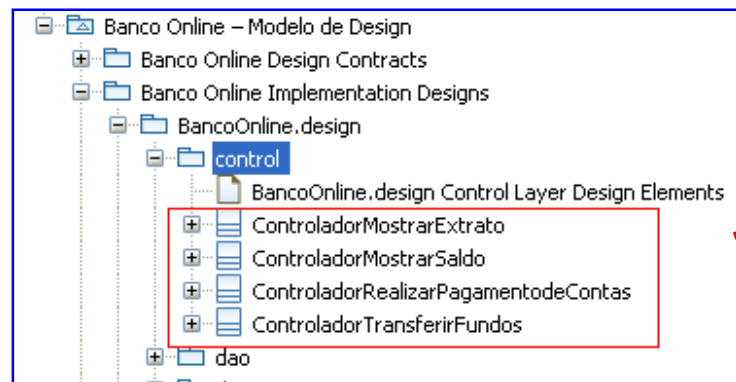
Laboratório ENGSOFT

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
- Repita a mesma coisa para as demais classes:



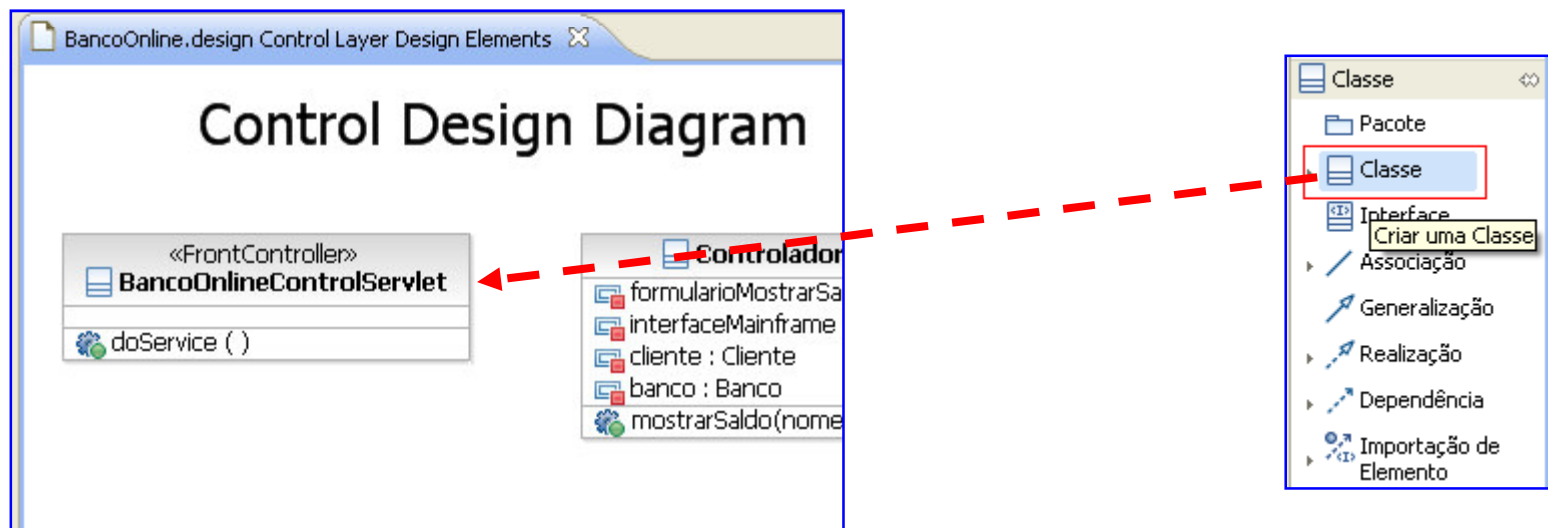
Laboratório ENGSOFT

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
- Copiar as 4 classes para o diagrama “BancoOnline.design Control Layer Design Elements”.



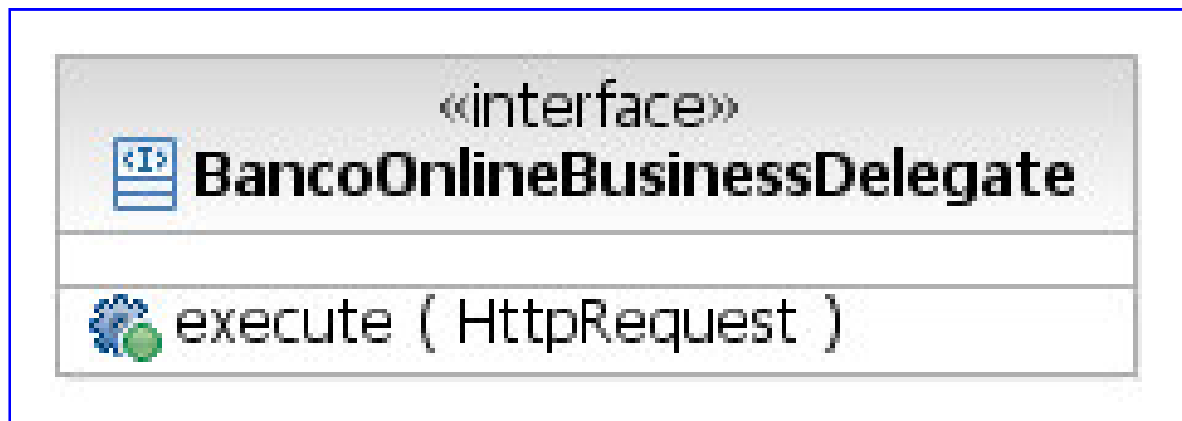
Laboratório ENGSOFT

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
- Vamos criar a classe controladora chamada “BancoOnlineControlServlet”. Clicar na paleta em “Classe” e arraste para o diagrama. Crie uma operação chamada “doService()”. Crie o esteriótipo chamado “FrontController”.



Laboratório ENGSO

- Estruturando a aplicação.
- Aplicação de alguns patterns para arrumar a camada de apresentação.
- Criando o “Business Delegate” (Command Pattern).
- Clicar na paleta e escolher “interface”. Arrastar até o diagrama e colocar o nome “IBancoOnlineBusinessDelegate”. Criar a operação “execute(HttpRequest)”.
- Criar uma agregação entre “BancoOnlineControlServlet” e “IBancoOnlineBusinessDelegate”.





Laboratório ENGSO

- Estruturando a aplicação.
- Criando as visões.
 - Clicar na paleta e escolher “Classe”. Dar o nome: “MostrarSaldoCommand” e o esteriótipo “Servlet”.
 - Clicar na paleta e escolher “Classe”. Dar o nome: “MostrarExtratoCommand” e o esteriótipo “Servlet”.
 - Clicar na paleta e escolher “Classe”. Dar o nome: “RealizarPagamentodeContasCommand” e o esteriótipo “Servlet”.
 - Clicar na paleta e escolher “Classe”. Dar o nome: “TransferirFundosCommand” e o esteriótipo “Servlet”.
 - Criar um relacionamento de dependência entre “BancoOnlineBusinessDelegate” e cada classe acima criada.

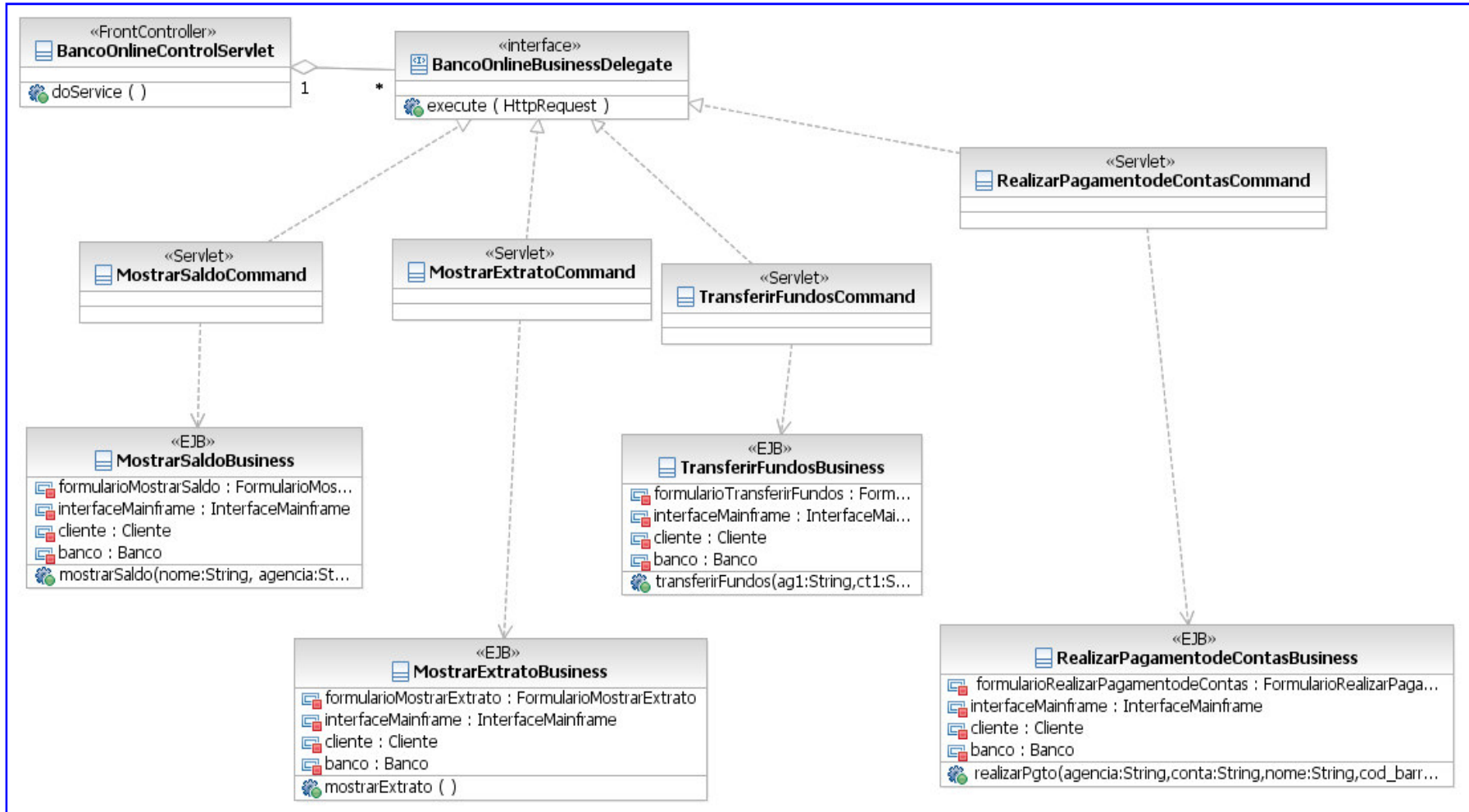


Laboratório ENGSO

- Estruturando a aplicação.
- Criando as visões.
 - Altere o nome da classe “ControladorMostrarSaldo” para “MostrarSaldoBusiness”. Coloque o esteriótipo “EJB”.
 - Aplique uma dependência entre “MostrarSaldoCommand” e “MostrarSaldoBusiness”.
 - Altere o nome da classe “ControladorMostrarExtrato” para “MostrarExtratoBusiness”. Coloque o esteriótipo “EJB”.
 - Aplique uma dependência entre “MostrarExtratoCommand” e “MostrarExtratoBusiness”.
 - Altere o nome da classe “ControladorRealizarPagamentodeContas” para “RealizarPagamentodeContasBusiness”. Coloque o esteriótipo “EJB”.
 - Aplique uma dependência entre “RealizarPagamentodeContasCommand” e “RealizarPagamentodeContasBusiness”.
 - Altere o nome da classe “ControladorTransferirFundos” para “TransferirFundosBusiness”. Coloque o esteriótipo “EJB”.
 - Aplique uma dependência entre “TransferirFundosCommand” e “TransferirFundosBusiness”.

Laboratório ENGSO

- Estruturando a aplicação.
- O Resultado:



Laboratório ENGSO

- Estruturando a aplicação.
 - Criando as visões: As seguintes classes de análise serão tratadas como componentes visuais (Arquivos .JSP)

UML Component Diagram for **FormularioMostrarSaldo**. The component is labeled «Limite» and has two empty rectangular slots for required interfaces.

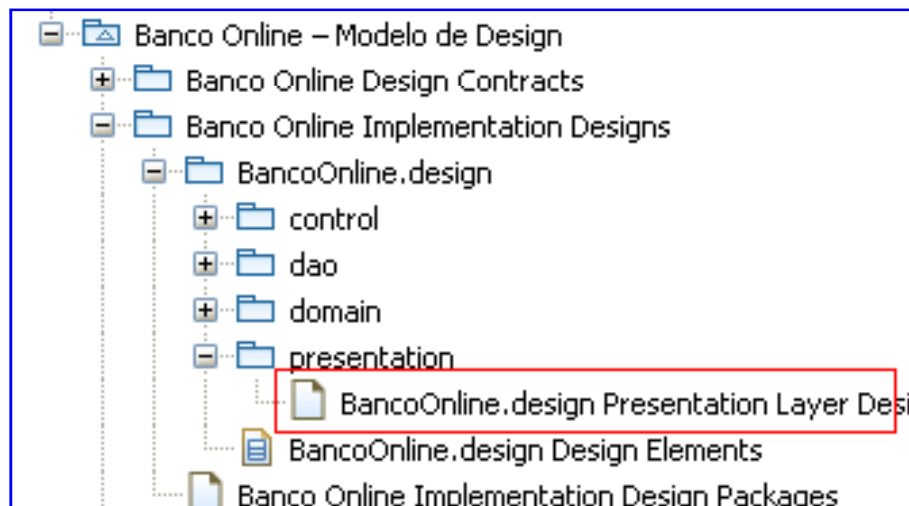
UML Component Diagram for **FormularioMostrarExtrato**. The component is labeled «Limite» and has two empty rectangular slots for required interfaces.

UML Component Diagram for **FormularioRealizarPagamentodeContas**. The component is labeled «Limite» and has two empty rectangular slots for required interfaces.

UML Component Diagram for **FormularioTransferirFundos**. The component is labeled «Limite» and has two empty rectangular slots for required interfaces.

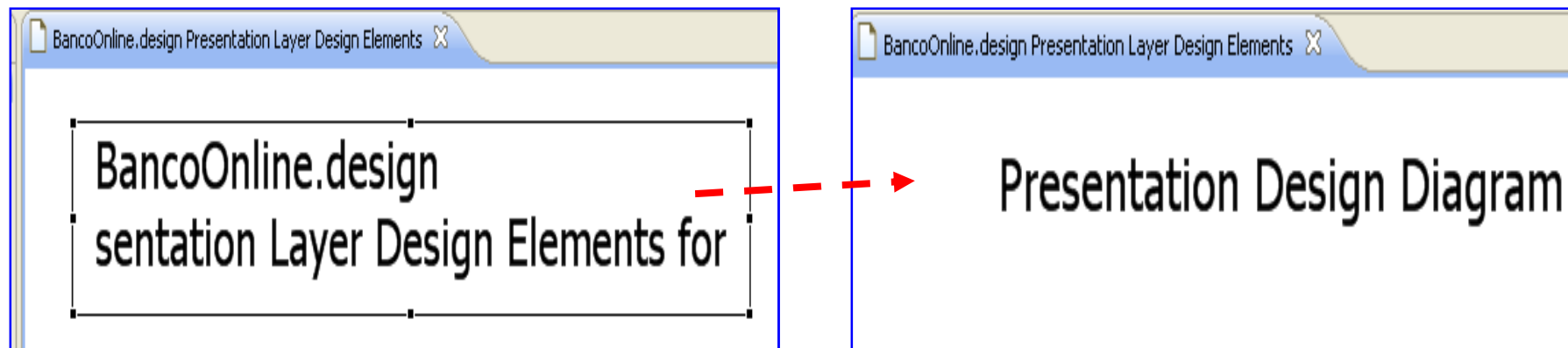
Laboratório ENGSOFT

- Estruturando a aplicação.
- Criando as visões. Utilização do pacote “presentation”.
 - Expandir “Banco Online – Modelo de Design”, em “Banco Online Implementation Designs”, abrir o diagrama “BancoOnline.design Presentation Layer Design Elements”.



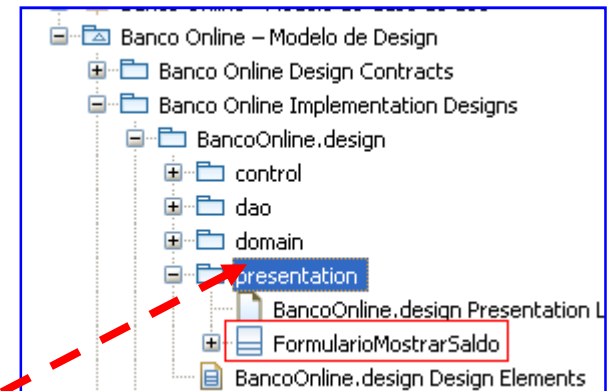
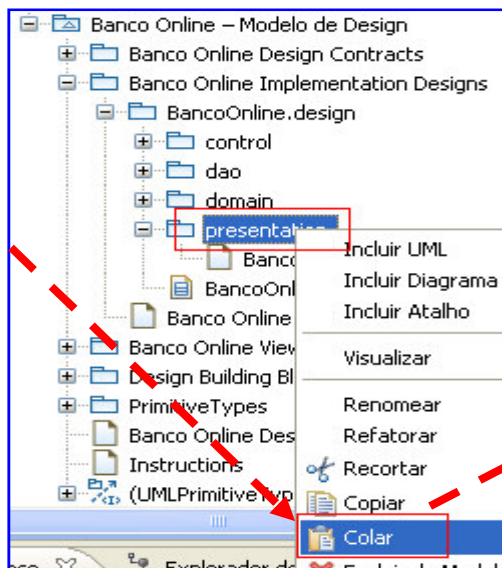
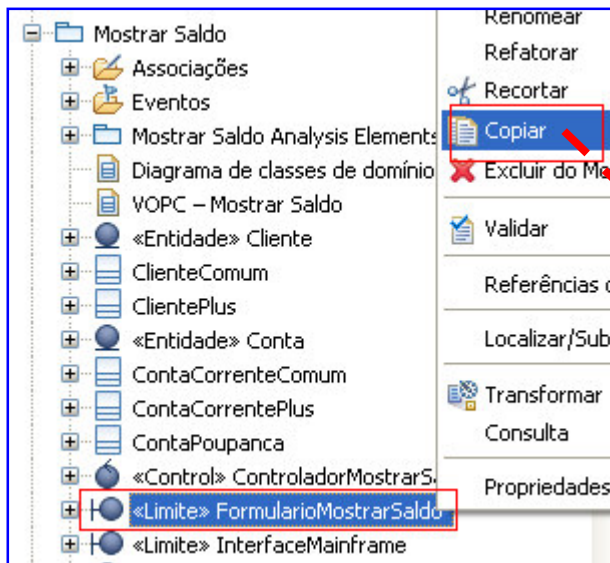
Laboratório ENGSOE

- Estruturando a aplicação.
- Criando as visões.
 - Abra o diagrama “BancoOnline.design Presentation Layer Design Elements”.
 - Renomear nome do diagrama “BancoOnline.design Presentation Layer Design Elements” para “Presentation Design Diagram”.Dê um “F2” em cima do “label”.



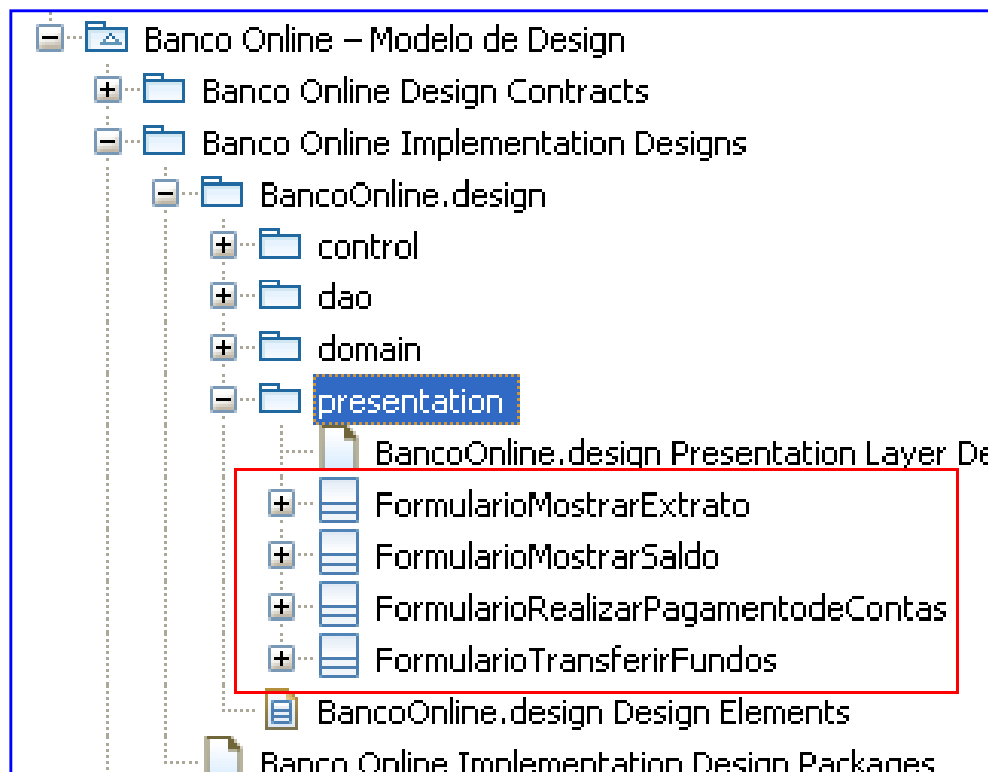
Laboratório ENGSOFT

- Estruturando a aplicação.
- Criando as visões.
- Abra o modelo “Banco Online – Modelo de Análise”->”Mostrar Saldo”. Clique com o botão direito em ” FormularioMostrarSaldo” e escolha “Copiar”.
- Abra o modelo “Banco Online – Modelo de Design”->” Banco Online Implementation Designs”->” BancoOnline.design”->”presentation”. Clique com o botão direito em “control” e escolha “Colar”.



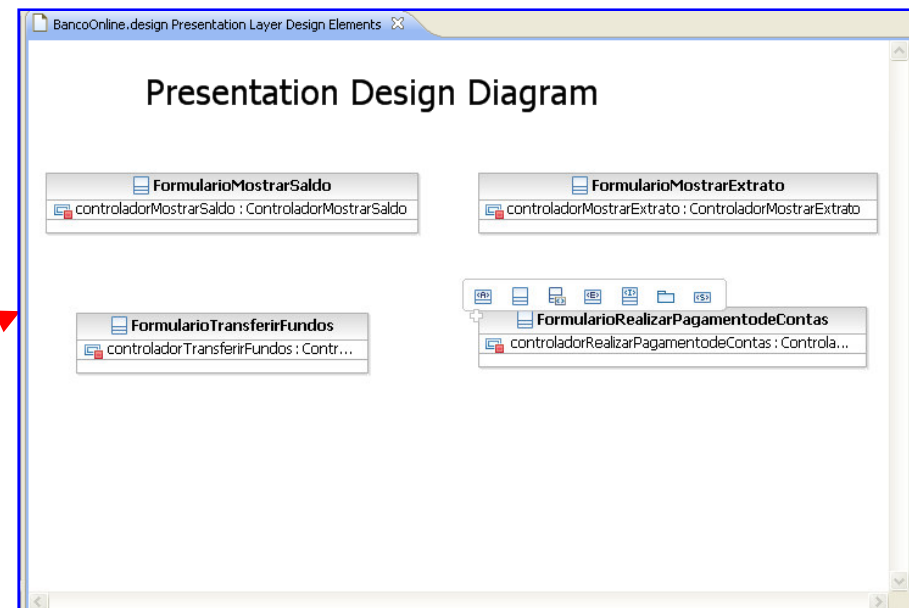
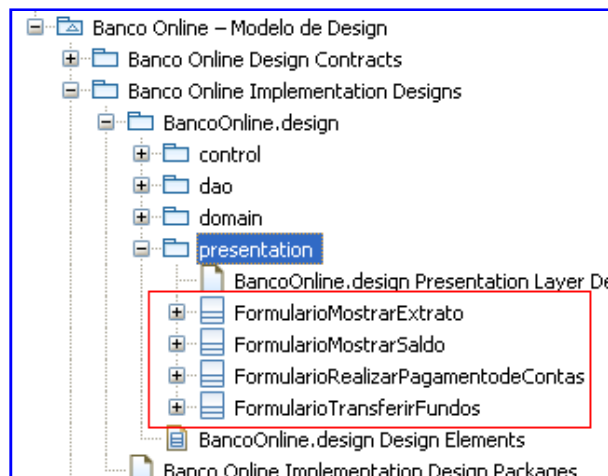
Laboratório ENGSOFT

- Estruturando a aplicação.
- Criando as visões.
- Repita a mesma coisa para as demais classes:



Laboratório ENGSOFT

- Estruturando a aplicação.
- Criando as visões.
- Copiar as 4 classes para o diagrama “BancoOnline.design Presentation Layer Design Elements”.



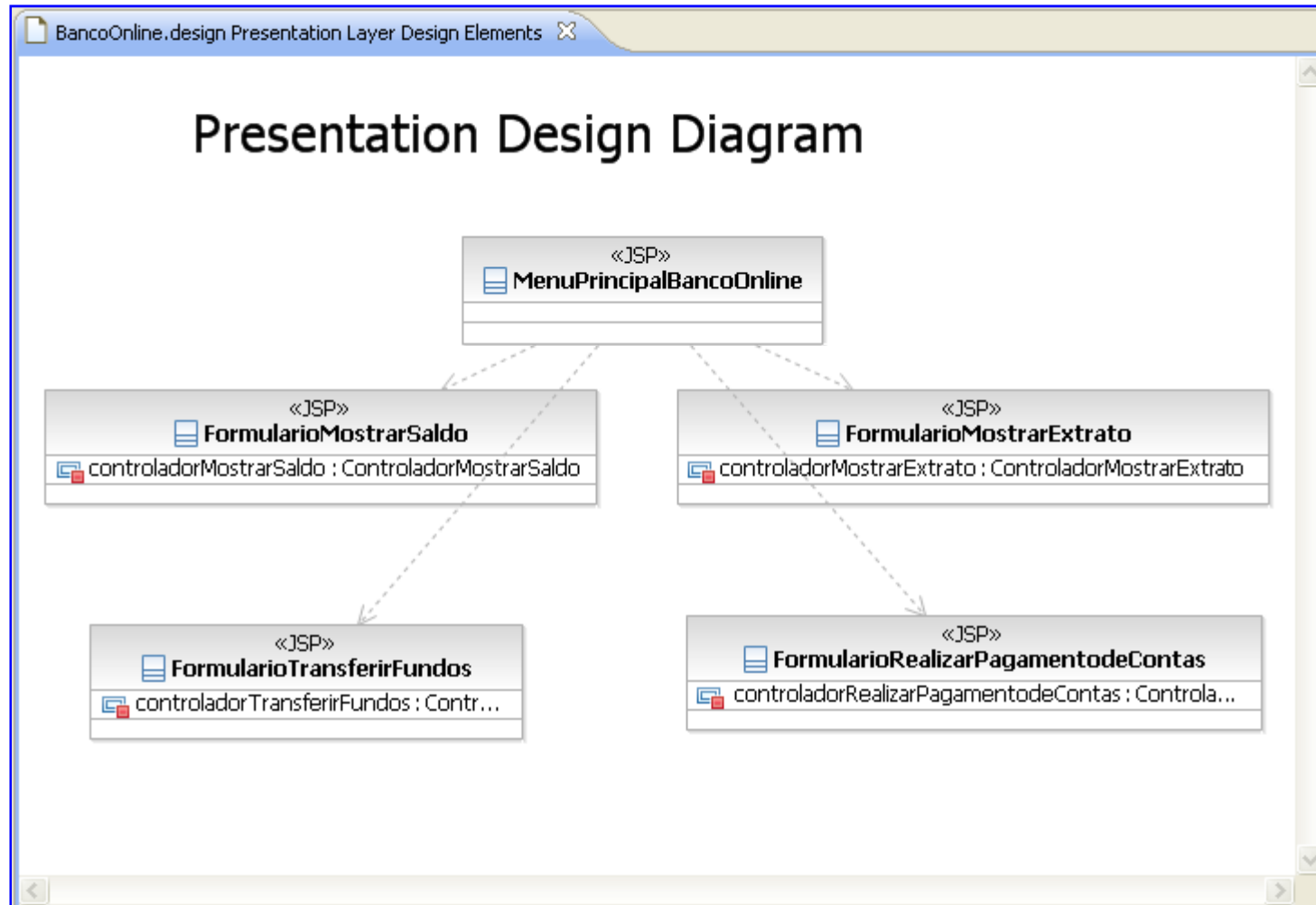


Laboratório ENGSO

- Estruturando a aplicação.
- Criando as visões.
 - Clicar na paleta e escolher “Classe”. Dar o nome: “MenuPrincipalBancoOnline” e o esteriótipo “JSP”.
 - Clicar em “FormularioMostrarSaldo” e colocar o esteriótipo “JSP”.
 - Clicar em “FormularioMostrarExtrato” e colocar o esteriótipo “JSP”.
 - Clicar em “FormularioTransferirFundos” e colocar o esteriótipo “JSP”.
 - Clicar em “FormularioRealizarPagamentodeContas” e colocar o esteriótipo “JSP”.
 - Criar um relacionamento de dependência entre “MenuPrincipalBancoOnline” e os demais componentes (.jsp). (Ver desenho e utilizar a seta “dependência” da paleta “classe”.

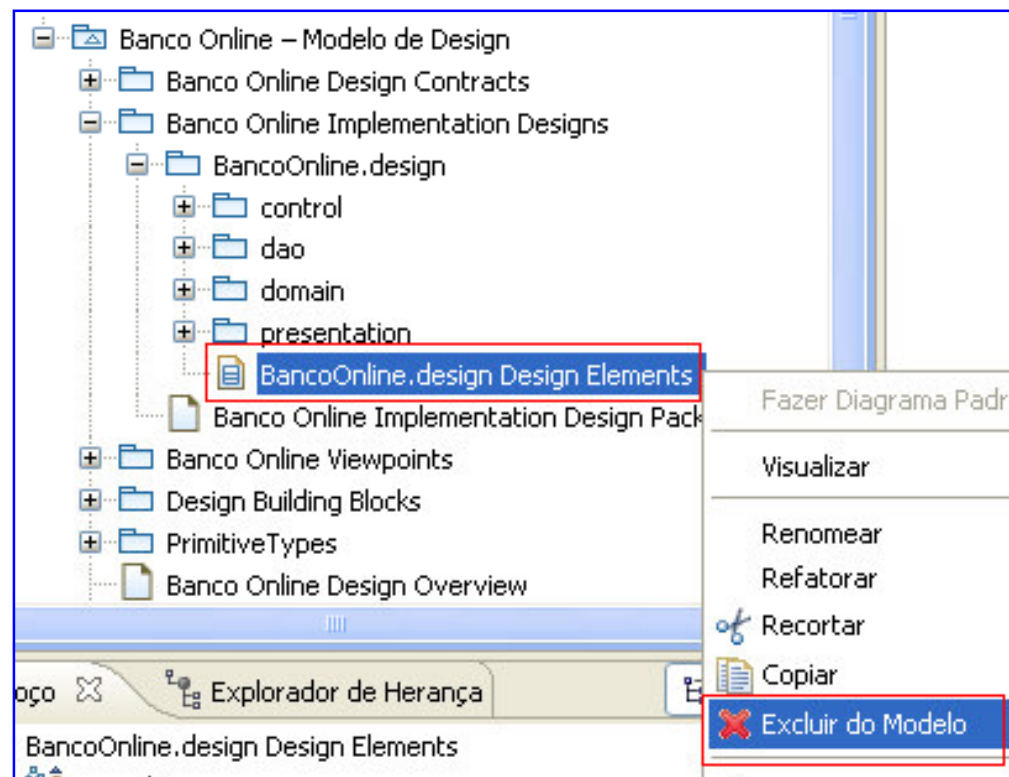
Laboratório ENGSO

- Criando o diagrama de pacotes da camada de Apresentação.
- Criando as visões: Resultado Final



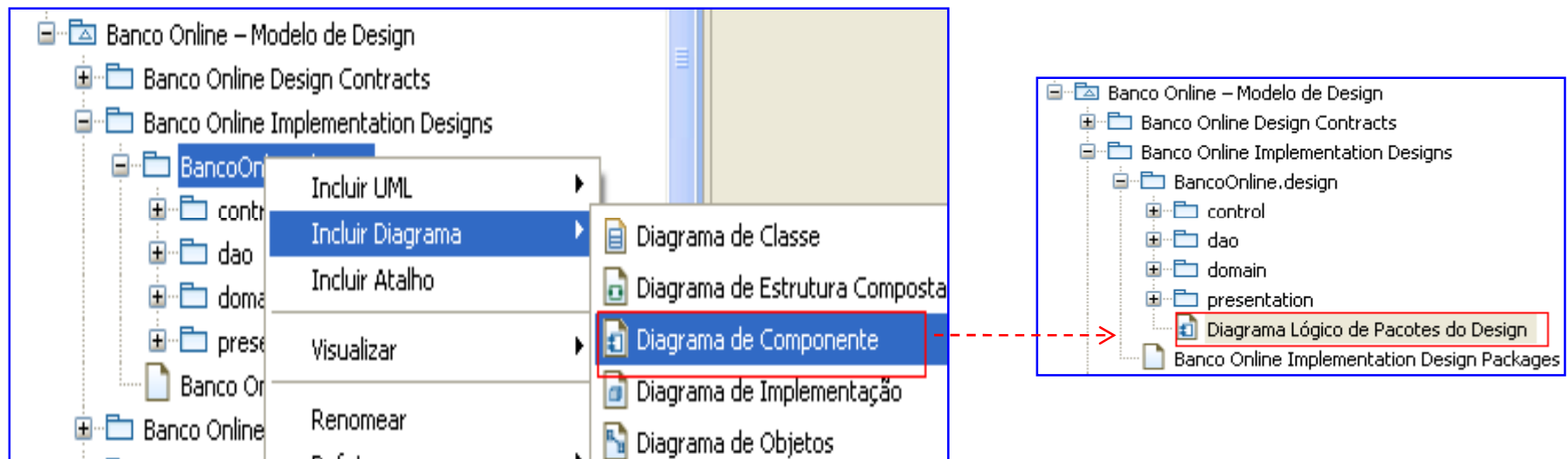
Laboratório ENGSOFT

- Criando o diagrama de Pacotes:
- Excluir o diagrama “BancoOnline.design Design Elements”.



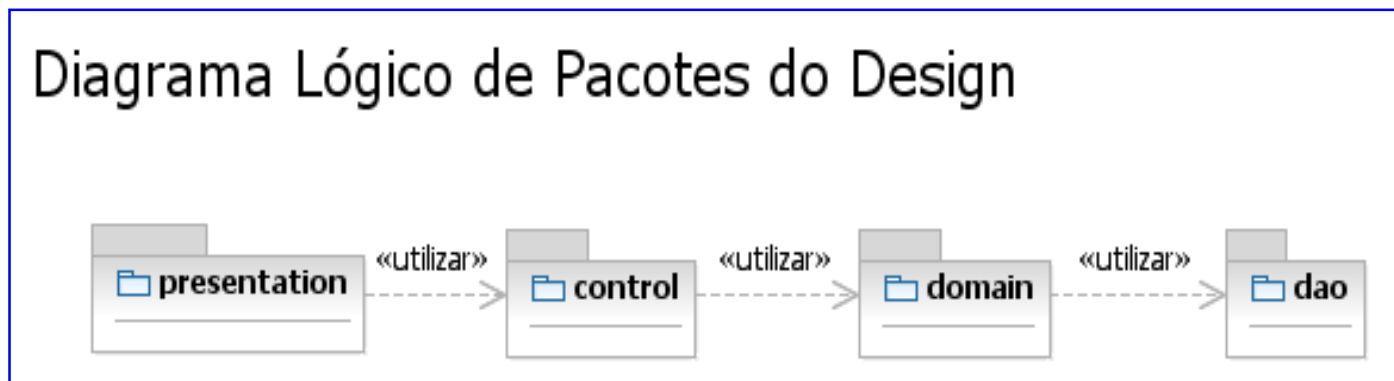
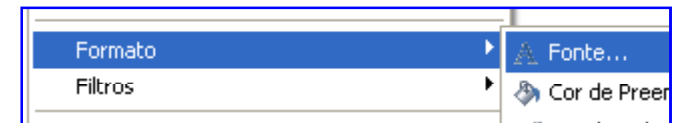
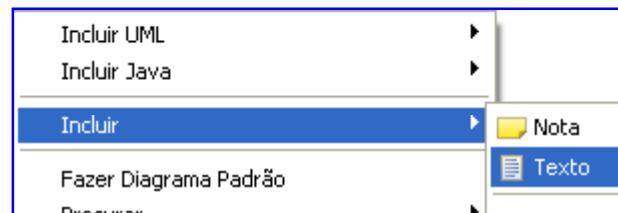
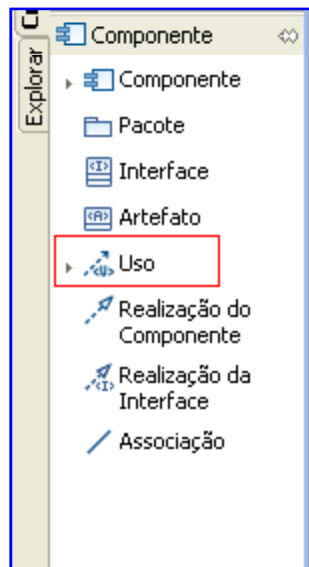
Laboratório ENGSOFT

- Criando o diagrama de Pacotes:
- Clicar em “Incluir Diagrama”-> “Diagrama de Componente” e crie o diagrama chamado “Diagrama Lógico de Pacotes do Design”.



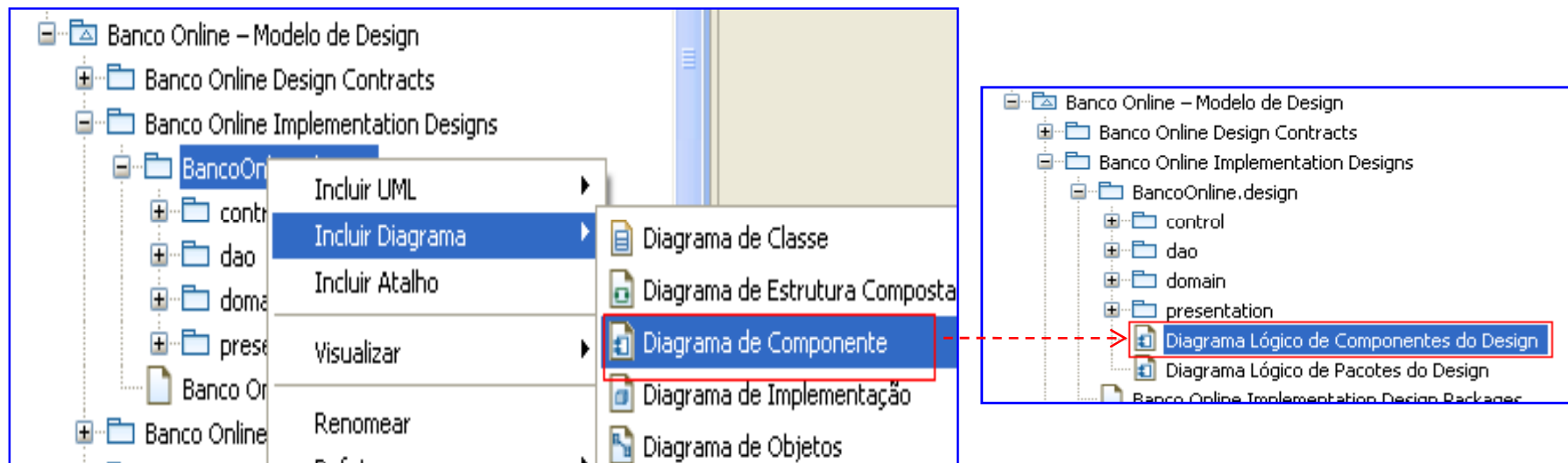
Laboratório ENGSOFT

- Criando o diagrama de Pacotes:
- Clique em cada pacote criado e arraste para o diagrama.
- Crie uma dependência (escolher na paleta a dependência “Uso”) entre os pacotes conforme diagrama abaixo.
- Crie um texto e coloque o nome: “Diagrama Lógico de Pacotes do Design” com o tamanho “18”.



Laboratório ENGSOFT

- Criando o diagrama de Componentes:
- Clicar em “Incluir Diagrama”-> “Diagrama de Componente” e crie o diagrama chamado “Diagrama Lógico de Componentes do Design”.



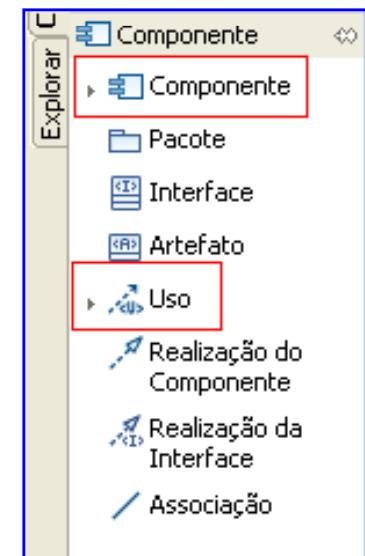
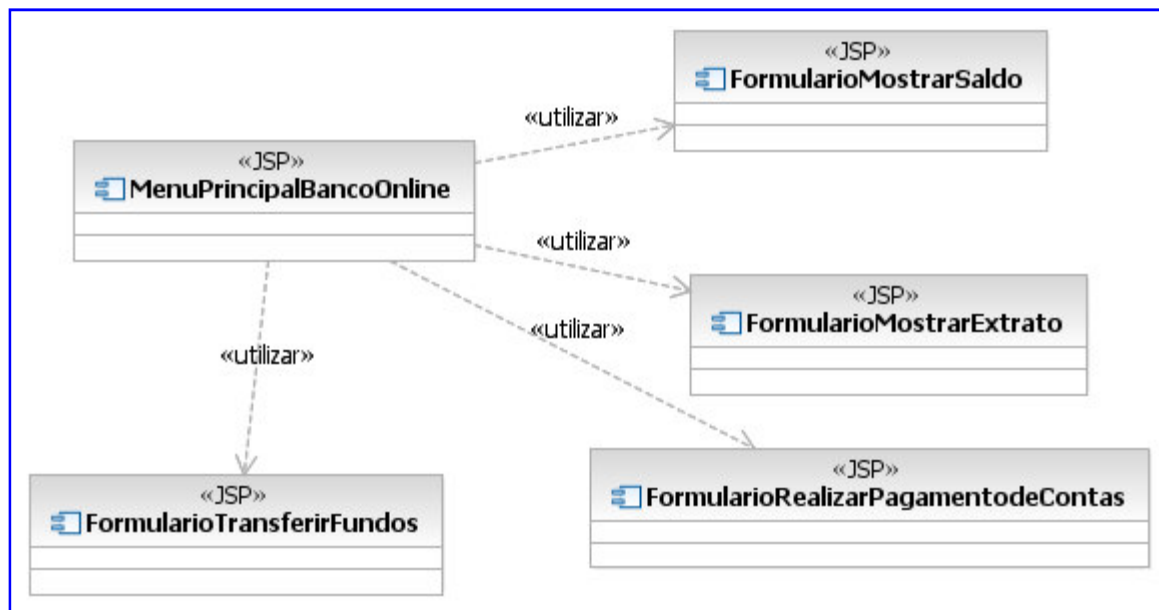


Laboratório ENGSO

- Criando o diagrama de Componentes:
- O diagrama de componentes abstrai que tipo de componentes teremos. Um pacote representa algo “físico” (Um diretório contendo vários elementos). Um componente representa algo “lógico” (Pode ser formado por 1 ou mais pacotes, ou várias classes).
- Dessa forma, evidenciaremos apenas os principais componentes do sistema e faremos as dependências entre os mesmos.
- Para isso, iremos criar um componente para cada classe que o representa, as quais:
 - **Pacote presentation:** MenuPrincipalBancoOnline, FormulárioMostrarSaldo, FormulárioMostrarExtrato, FormulárioTransferirFundos e FormulárioRealizarPagamentodeContas.
 - **Pacote control:** BancoOnlineControlServlet, BancoOnlineBusinessDelegate(essa interface representa todas as classes referenciadas).
 - **Pacote domain:** Pegaremos emprestado as classes “Business” do pacote control e vamos criar um componente para cada uma.
 - **Pacote dao:** IBOdaoFactory.

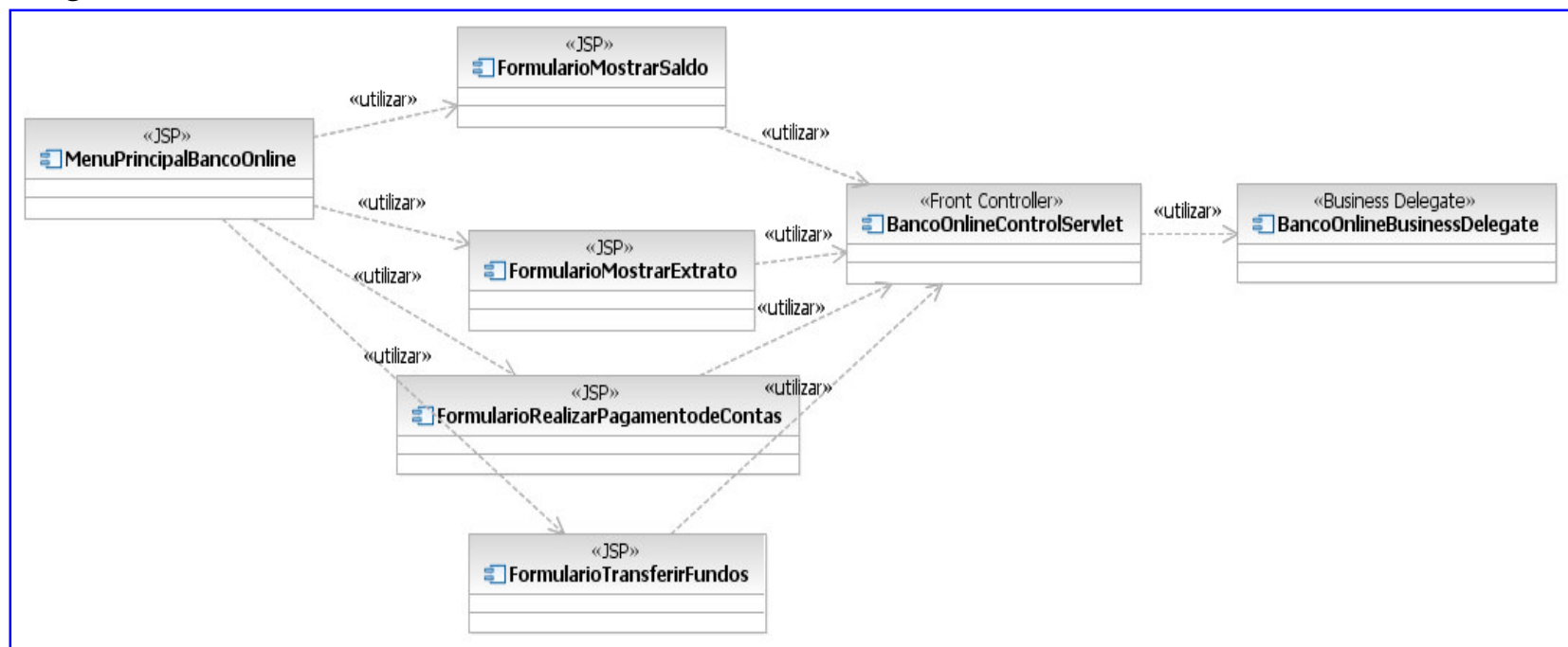
Laboratório ENGSOFT

- Criando o diagrama de Componentes:
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “MenuPrincipalBancoOnline”.
- Repetir para as classes: FormularioMostrarSaldo, FormularioMostrarExtrato, FormularioTransferirFundos e FormularioRealizarPagamentodeContas.
- Coloque o esteriótipo para cada classe :”JSP”.
- Crie dependências entre os componentes criados conforme a lógica já definida pelo diagrama de classes.



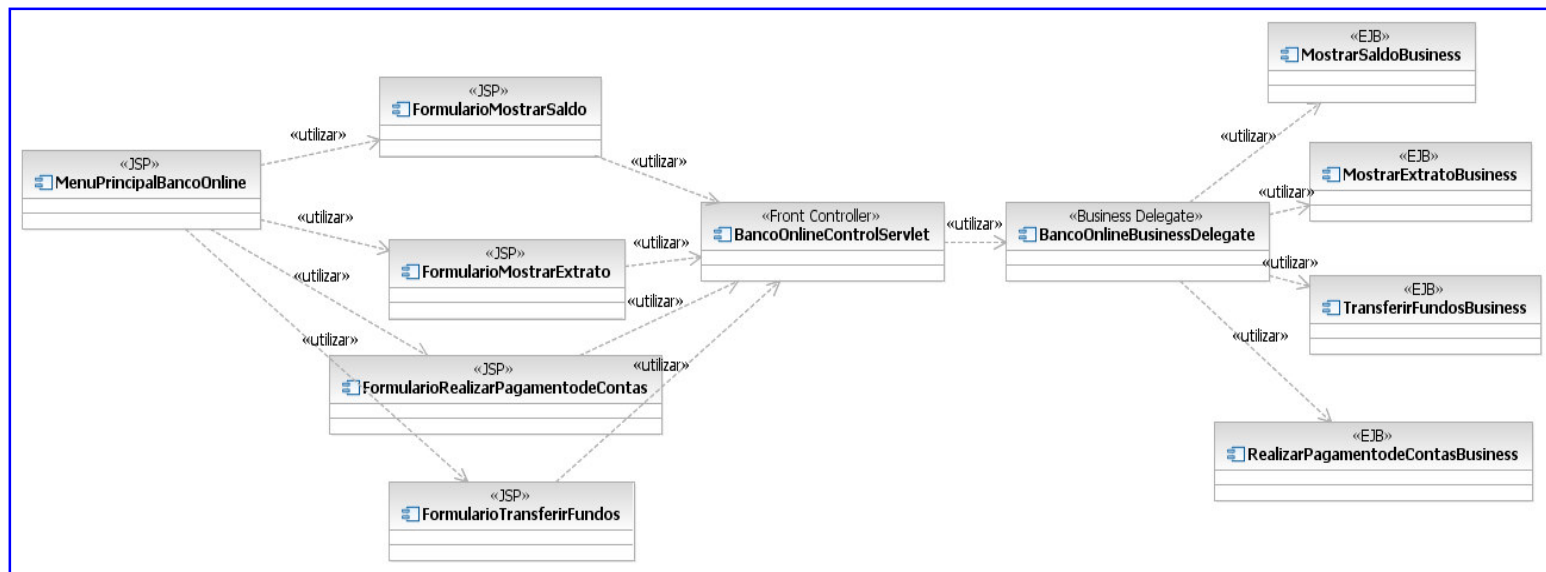
Laboratório ENGSOFT

- Criando o diagrama de Componentes:
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “BancoOnlineControlServlet”. Coloque o esteriótipo “Front Controller”.
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “BancoOnlineBusinessDelegate”. Coloque o esteriótipo “Business Delegate”.
- Crie dependências entre os componentes criados conforme a lógica já definida pelo diagrama de classes.



Laboratório ENGSO

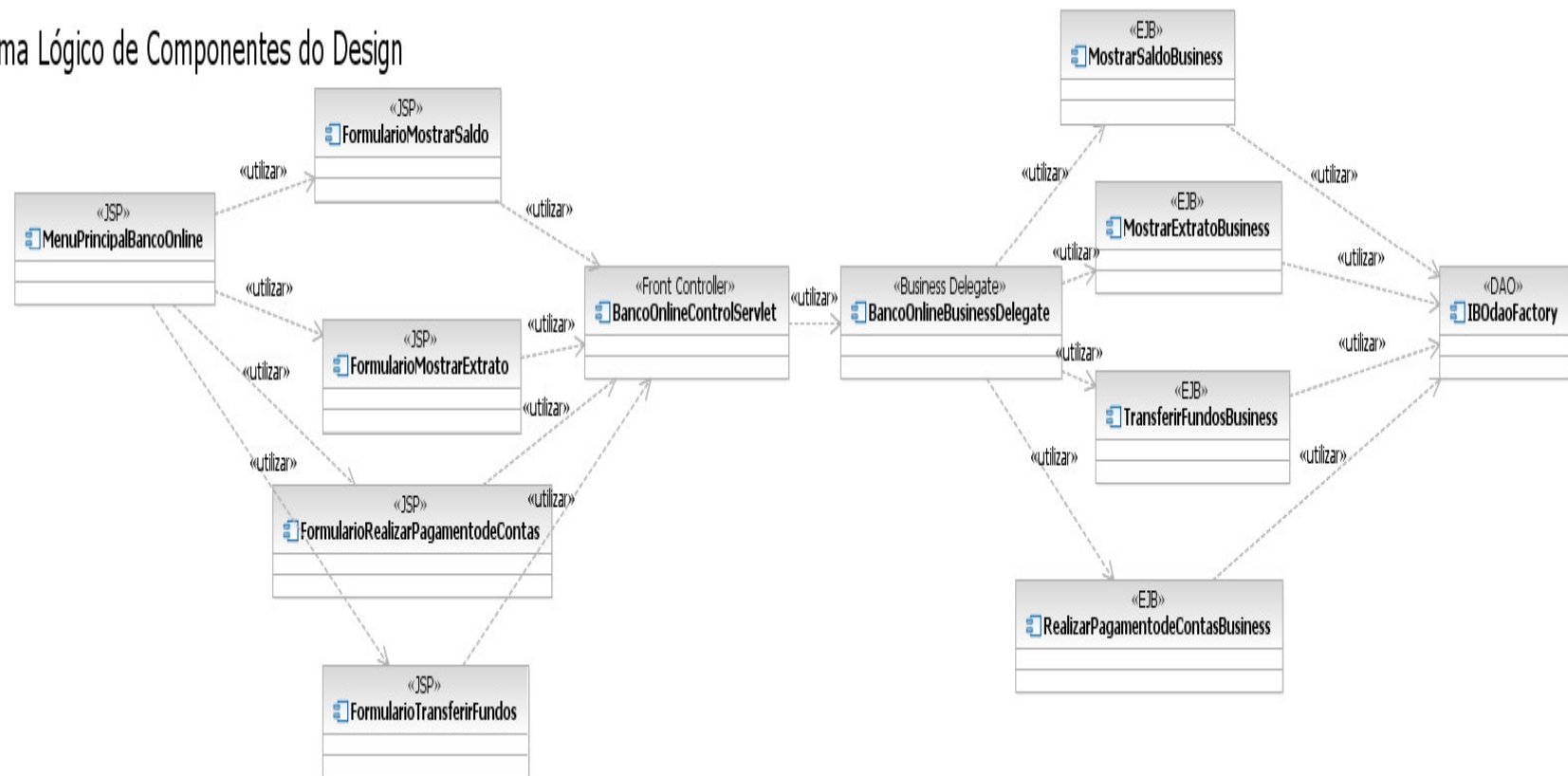
- Criando o diagrama de Componentes:
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “MostrarSaldoBusiness”. Coloque o esteriótipo “EJB”.
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “MostrarExtratoBusiness”. Coloque o esteriótipo “EJB”.
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “RealizarPagamentodeContasBusiness”. Coloque o esteriótipo “EJB”.
- Clicar na paleta e escolher “Componente” e arrastar até o diagrama. Colocar o nome “TransferirFundosBusiness”. Coloque o esteriótipo “EJB”.
- Crie dependências entre os componentes criados conforme a lógica já definida pelo diagrama de classes.



Laboratório ENGSO

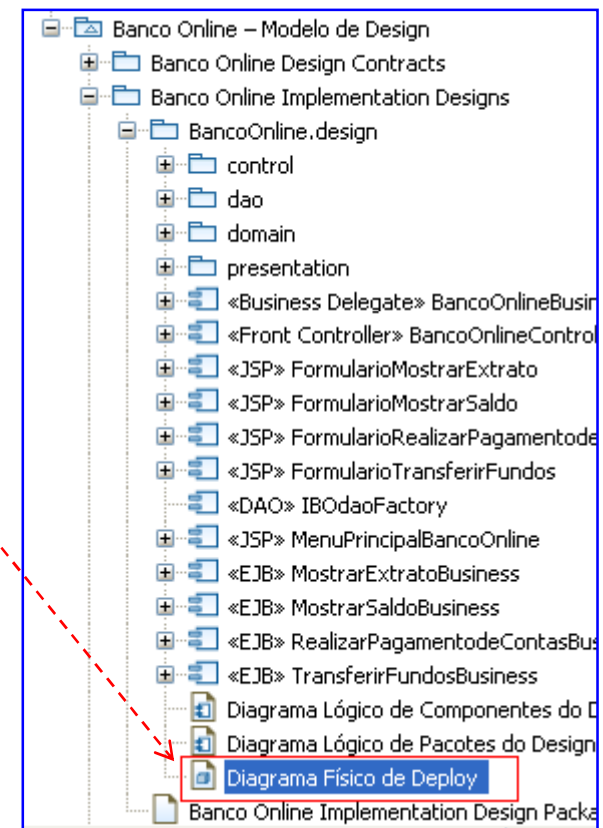
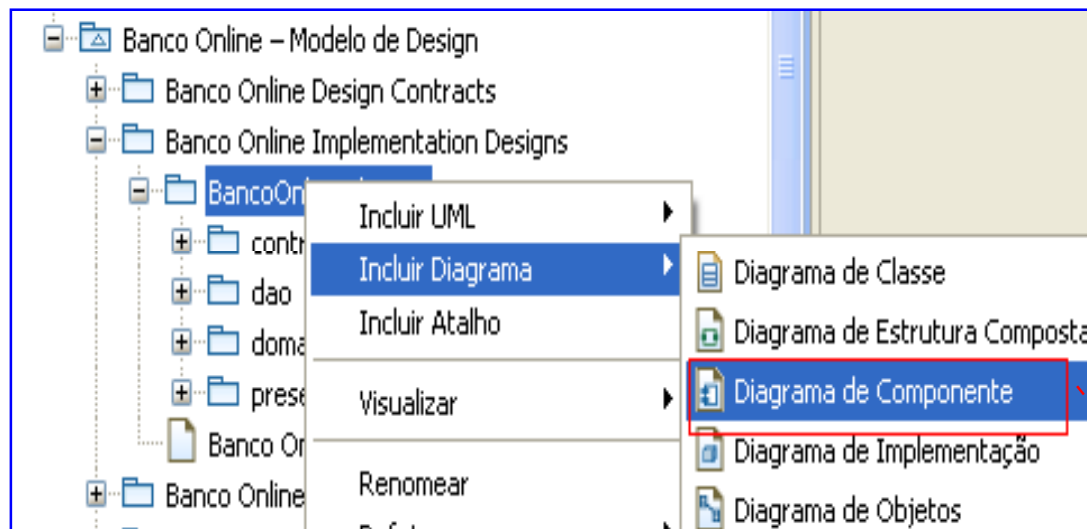
- Criando o diagrama de Componentes:
- Por último, criar o componente “IBOdaoFactory”, coloque o esteriótipo “DAO”.
- Finalize as dependências e coloque o texto: “Diagrama Lógico de Componentes do Design”.

Diagrama Lógico de Componentes do Design



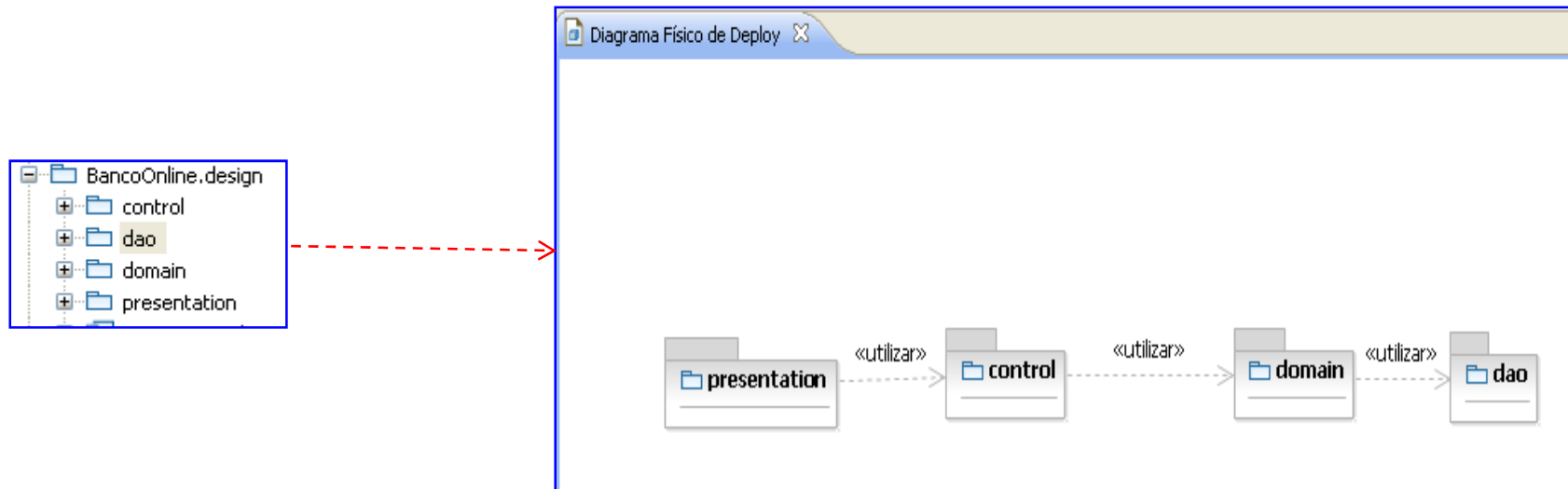
Laboratório ENGSOFT

- Criando o diagrama de Deploy (ou Físico):
- Clicar em “Incluir Diagrama”-> “Diagrama de Componente” e crie o diagrama chamado “Diagrama Físico de Deploy”.



Laboratório ENGSOFT

- Criando o diagrama de Deploy (ou Físico):
- Abra o diagrama “Diagrama Físico de Deploy”.
- Arrastar os pacotes para o diagrama “Diagrama Físico de Deploy” conforme abaixo:





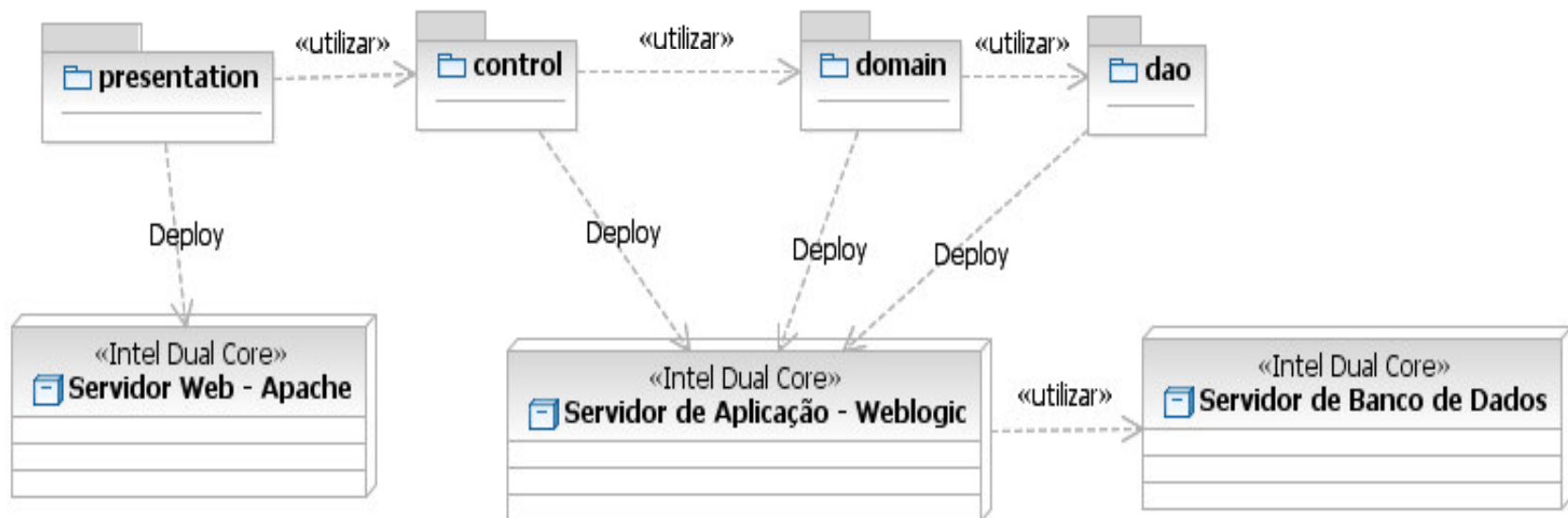
Laboratório ENGSO

- Criando o diagrama de Deploy (ou Físico):
 - Clicar na paleta em “Nó” e colocar no diagrama. Colocar no nome “Servidor Web - Apache” e esteriótipo “Intel Dual Core”.
 - Clicar na paleta em “Nó” e colocar no diagrama. Colocar no nome “Servidor de Aplicação - Weblogic” e esteriótipo “Intel Dual Core”.
 - Clicar na paleta em “Nó” e colocar no diagrama. Colocar no nome “Servidor de Banco de Dados” e esteriótipo “Intel Dual Core”.
 - Crie as seguintes dependências do tipo “Dependência” entre cada nó e cada pacote conforme figura abaixo. Coloque o label “Deploy” em cada uma.
 - Crie uma dependência do tipo “Uso” entre “Servidor de Aplicação – Weblogic” e “Servidor de Banco de Dados”.
 - Coloque o texto “Diagrama Físico de Deploy”.

Laboratório ENGSOFT

- Criando o diagrama de Deploy (ou Físico):
- Diagrama final:

Diagrama Físico de Deploy





Laboratório ENGSO

- O que você aprendeu até aqui?
- Porque precisamos refatorar no design?
- Qual é a diferença entre Componentes e Pacotes?
- Percebeu o beneficio de design patterns aplicados à aplicação?