



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

ESCOLA DE INFORMÁTICA APLICADA

DOCUMENTAÇÃO DO TRABALHO 2

Leonardo dos Santos Silva

Wilian Pereira da Silva Santos

Professor

Alexandre Correa

Disciplina

Programação Modular

1. Visão Geral

O programa é responsável por cadastrar e manter em um banco de dados os dados relacionados a um Pesquisador de uma produção bibliográfica, ou seja, o programa faz inclusão, alteração, exclusão e consulta de dados de um Pesquisador. Além disso, realiza consulta e emite relatório de uma Publicação utilizando dados pré-cadastrados da mesma no banco de dados.

2. Descrição das operações

2.1. Pesquisador

- **Cadastrar Pesquisador**

Inserere um Pesquisador com os campos: Primeiro Nome, Nome do Meio, Último Nome, Logradouro, Tipo de Logradouro, Número, Complemento, Cidade, Estado, País e Afiliação.

- **Consultar Pesquisador**

Consulta um Pesquisador através do Primeiro Nome, Nome do Meio e/ou Último Nome.

- **Alterar Pesquisador**

Para alterar os campos de Pesquisador, o usuário utilizará primeiro a operação “Consultar Pesquisador” para consultar o mesmo pelo Primeiro Nome, Nome do Meio e/ou Último Nome. Após o retorno da busca, o usuário poderá alterar os campos Nome, Nome do meio, Último Nome, Tipo de Logradouro, Logradouro, Número e Complemento.

- **Excluir Pesquisador**

Para excluir um Pesquisador, o usuário utilizará primeiro a operação “Consultar Pesquisador” para consultar o mesmo pelo Primeiro Nome, Nome do Meio e/ou Último Nome. Após o retorno da busca, o usuário poderá excluir o Pesquisador solicitado.

- **Listar Pesquisadores**

Lista todos os Pesquisadores cadastrados no banco de dados. O programa exibe o Primeiro Nome, Nome do Meio e Último Nome de cada Pesquisador.

2.2. Publicação

- **Consultar Publicação**

Consulta uma Publicação através do Título, Classificação Qualis, Evento e Quantidade mínima que a produção bibliográfica influencia outras produções.

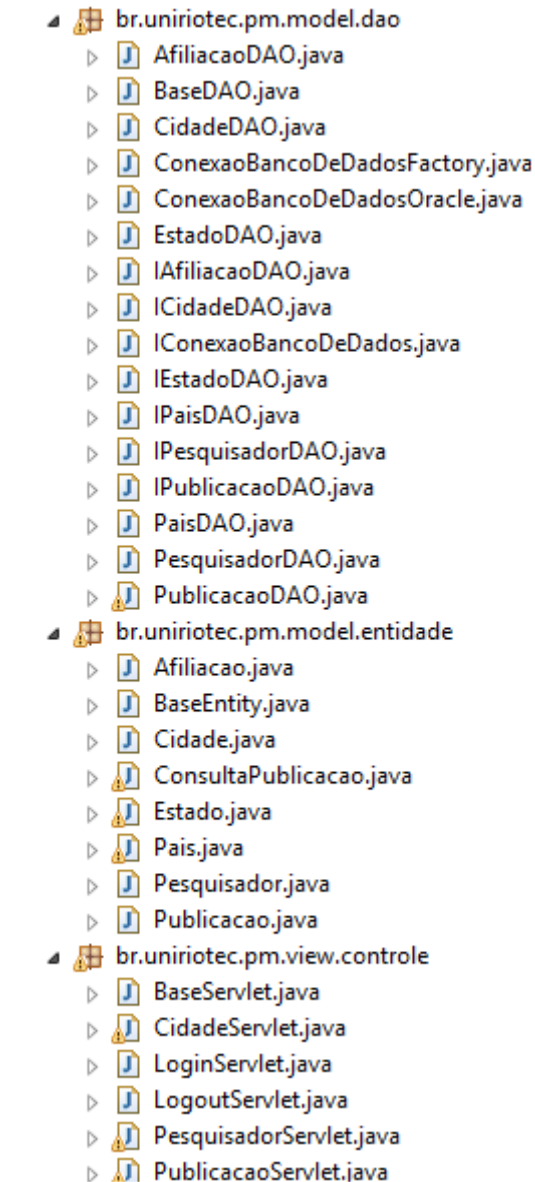
- **Relatório da Publicação**

Lista todas as Publicações que atenderem aos critérios da consulta feita pela operação “Consultar Publicação”. Os campos listados serão: Título, Quantidade de produções influenciadas, Classificação Qualis, Quantidade de Pesquisadores, Nome do Primeiro Pesquisador, Cidade da Afiliação do Pesquisador e Nome de Citação.

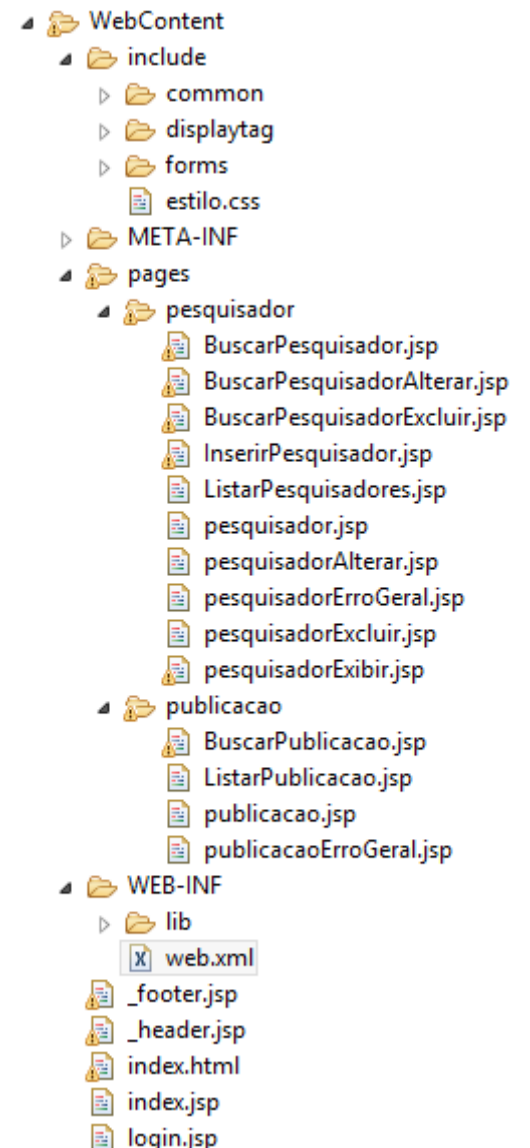
3. Banco de dados

O programa utiliza o Oracle como SGBD.

4. Estrutura



- br.uniriotec.pm.model.dao: pacote que contém as classes que representam os dados do programa, bem como o acesso ao banco de dados. Além disso, o pacote também possui as interfaces relacionadas às classes que as implementam.
- br.uniriotec.pm.model.entidade: pacote que contém as classes que representam os dados do programa de acordo com o modelo de dados.
- br.uniriotec.pm.view.controle: pacote que contém as classes que controlam as requisições relacionadas com os objetos de cada tipo de dado (Cidade, Pesquisador, Publicação, etc).



- Web Content: possui todo o conteúdo voltado para a web, como arquivos .jsp, .html, .css e figuras.
 - pages: possui as páginas da web referentes à Pesquisador e Publicação.
- WEB-INF:
 - web.xml: onde ocorre o mapeamento do Servlet com as urls e o mapeamento de outras páginas.

5. Interface gráfica

A interface gráfica do programa é web. Possui uma tela de login e senha, um menu com as principais funções do programa e está organizado da seguinte maneira:

- Pesquisador
 - Inserir
 - Alterar
 - Excluir
 - Consultar
 - Listar Todos
- Publicação
 - Relatório
 - Consultar

6. Arquitetura

6.1 Padrão de arquitetura MVC

Foi utilizado neste trabalho o padrão de arquitetura MVC (Model, View, Controller). A estrutura pode ser representada pelo esquema da Figura 1.

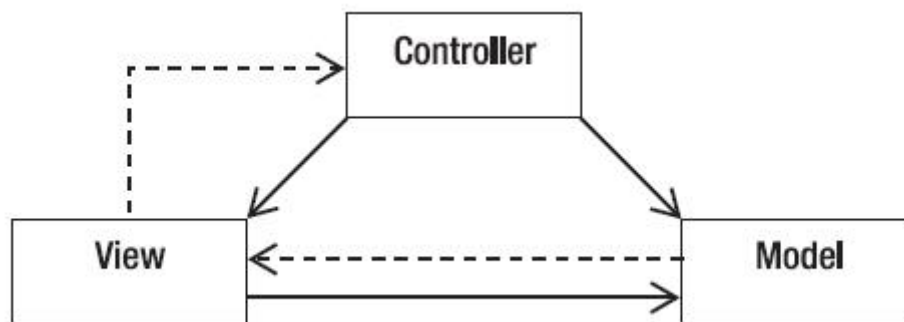


Figura 1 – Interações nos objetos MVC

No caso do trabalho, o controlador (Controller) é representado pelos servlets presentes no pacote `br.uniriotec.pm.view.controle`, e são responsáveis por interpretar as entradas do mouse ou do teclado enviadas pelo usuário em comandos que são enviados para o modelo (Model) e/ou para a visão (View) para que aconteçam as alterações necessárias.

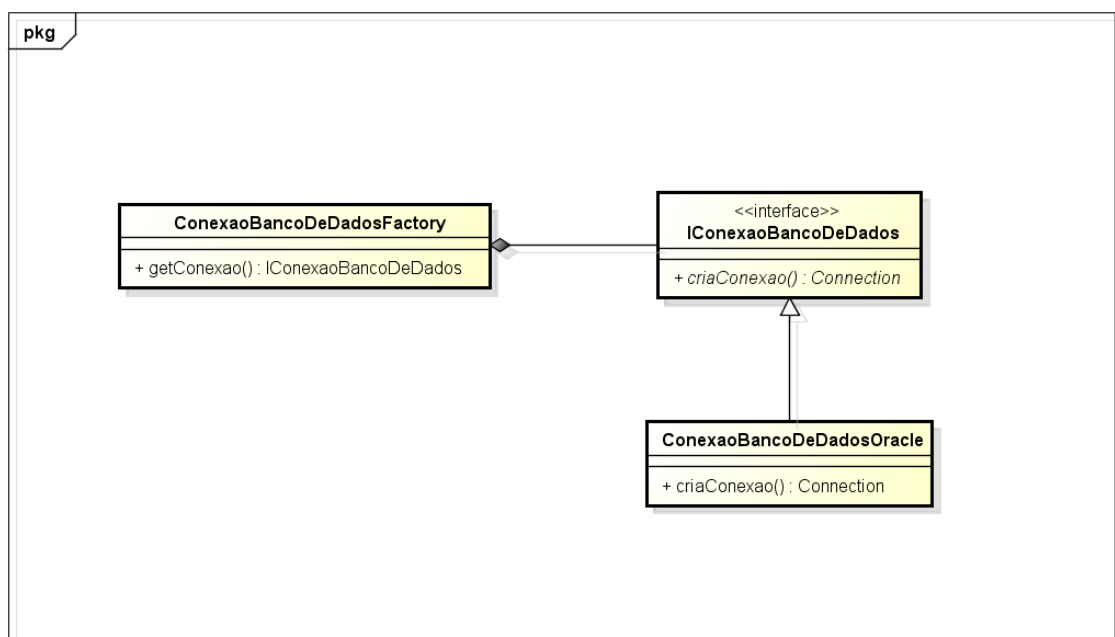
O modelo, que neste trabalho é representado pelas classes de dados e acesso ao mesmos, presentes no pacote `br.uniriotec.pm.modelo`, gerencia os elementos de dados, responde as perguntas sobre o seu estado e responde a instruções para mudar de estado.

Já a visão, representada por arquivos como `jsp`, `css`, `html` e presentes no diretório `Web-Content`, é responsável por apresentar as informações ao usuário. A visão recebe informações do controle e instruções do modelo, para, então, exibi-las. A visão também se comunica de volta com o modelo e o controlador para reportar o seu estado.

6.2 Padrão Factory

Foi utilizado o padrão Factory para fazer a conexão com o banco de dados. O padrão Factory permite a definição de uma classe para a criação de objetos e cada subclasse fica responsável por criar seus objetos.

Neste trabalho, utilizamos a classe `ConexaoBancoDeDadosOracle` que implementa a interface `IConexaoBancoDeDados`. A classe que representa o Factory, `ConexaoBancoDeDadosFactory`, implementa um método que retorna um novo objeto de `ConexaoBancoDeDadosOracle` (Figura 2). Nesse caso, só há uma implementação de banco de dados, porém, se tivesse mais de uma implementação, os objetos poderiam ser criados no Factory por meio de uma cláusula switch e passando um tipo por parâmetro.



powered by Astah

Figura 2 – Conexão com Banco de Dados utilizando Factory