

Backend Properties

Sistemas Distribuídos

Rui Raposo

Universidade da Beira Interior
Licenciatura em Engenharia Informática
Projeto
Doutora Professora Paula Prata

16 de Maio de 2020

Resumo

Java EE e, inclusive no *Java* ME.

Neste projeto são usados conceitos estudados no âmbito das aulas teóricas/práticas lecionadas nas últimas três semanas. Este documento é apenas **exclusivo** a elementos do **grupo treze**.

1 Introdução

Foi requisitada a criação de um sistema *web based* por parte da professora Paula Prata para avaliação da unidade curricular de sistemas distribuídos. Este, deve satisfazer inúmeros requisitos especificados no enunciado.

Um sistema *web based* funciona sempre, independentemente da linguagem, da mesma maneira, daí a pertinência deste documento ser a ajuda extra que possa ser dada a todos os leitores.

2 Ferramentas

As ferramentas utilizadas para o desenvolvimento deste projeto são o *Java Server Faces*, o *XHTML* e *JAVA DB*. Seguem-se três subsecções relativas às tecnologias explicitadas anteriormente.

2.1 *Java Server Faces*

Efetua a elaboração de *interfaces* de utilizadores para sistemas *web*, colocando componentes em um formulário e ligando-os a objetos *Java*.

2.2 XHTML

É uma reformulação da linguagem HTML baseada em XML. Combina as *tags* de marcação HTML com regras do XML.

2.3 *JAVA DB*

O *Java DB*, é uma base de dados cem por cento baseado em *Java*. Pode ser usado na plataforma *Java SE*,

3 *Backend*

De uma maneira concisa, podemos afirmar que o *backend* trata todo o processo que é desenvolvido no lado do servidor, desde a gestão de todos os dados fornecidos pelos utilizadores até ao suporte dado ao *frontend*, lado do cliente. Pode-se visualizar o esquema habitual de um sistema *web based* na figura a baixo.

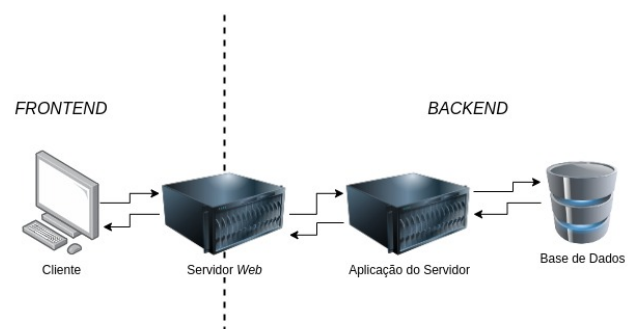


Figura 1: Esquema de um *web server*.

Nada melhor do que exemplificar todo este processo com base num exemplo, usando todas as ferramentas utilizadas no projeto.

3.1 Exemplo – Criação da Base de Dados e Tabelas

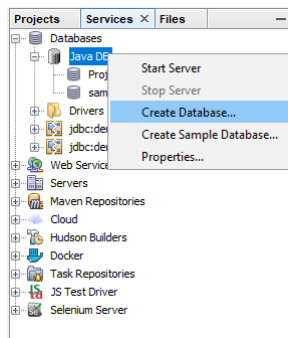


Figura 2: 1. Criação da base de dados

Após a inserção do *username* e da respetiva *password* irá ser visto algo deste género (nome da base de dados, neste caso exemplo, **Exemplo**).

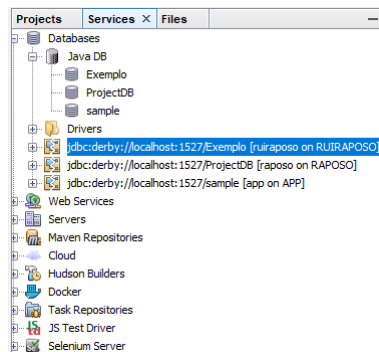


Figura 3: Visualização dos *services*

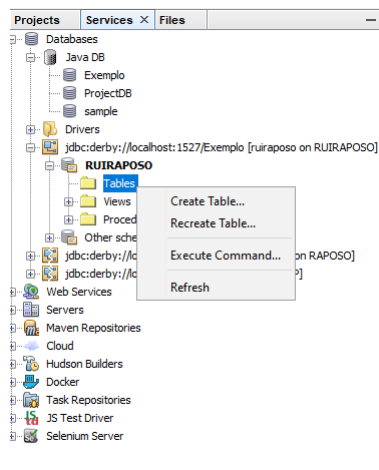


Figura 4: 2. Criação de Tabelas

3.2 Exemplo – Autenticação e Registo

Após a criação de todas as tabelas da referente base de dados, segue-se o desenvolvimento da aplicação em si. Seguidamente, será representado, esquematicamente, como deve operar o nosso sistema nas ações específicas de autenticação e de registo.

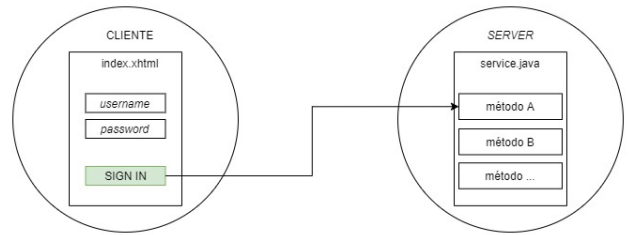


Figura 5: Visão global do sistema

Estando a trabalhar com o *Java Server Faces* seremos obrigados a utilizar ficheiros com extensão **XHTML**, esta linguagem é bastante parecida ao **HTML**, apenas é necessário verificar toda a sintaxe *online*.

Analizando de uma forma mais específica o lado do cliente, quando um cliente insere os dados na *input field* **username** e a *password* na outra, após este acionar o botão **sign in**, é **obrigatório** que estas estejam submetidas numa *form*. Segue-se uma série de normas que conduzem a uma boa prática da criação desta página.

1. Atribuição de uma *tag* do tipo *binding* a cada *input field*;
2. Associar uma variável a esse binding, algo do tipo `#var_name`;
3. No botão que queiramos que seja efetuado algum método, adicionar a *tag action*;
4. Associar um método do respetivo serviço/*bean* que queiramos que este seja despoletado, algo do género `#servico.methodA()`.

Note-se que, neste último ponto será necessário atribuímos um valor à classe para poder ser acedida nos ficheiros **XHTML**, tal como:
`@ManagedBean(name="servico")`.

Referências

Referências