

Trabalho 2 Tecnologias Web

Por: Carlos Palma 46520

Introdução

Foi proposto pelo professor José Saias responsável pela cadeira de Tecnologias Web do curso de Engenharia Informática da Universidade de Évora a elaboração de uma nova interface web para gerir os anúncios de arrendamento e as solicitações de procura. Neste segundo trabalho teria que desenvolver toda a aplicação deste a interface web às funcionalidade server-side de forma a conseguir páginas com os conteúdos necessários, apresentação com layout fluido e comportamento responsivo, armazenamento persistente, em PostgreSQL, tolerância a quebras de conectividade com o servidor, mantendo interação amigável e segurança, evitando vulnerabilidades comuns e observando regras de autorização de acesso.

Desenvolvimento

Para desenvolver a aplicação web optei por utilizar páginas JSP para apresentar o conteúdo, Spring JPA para autenticação com Data Access Object (DAO), controlador (MVC) Spring e Postgres para armazenamento persistente de dados. A apresentação do conteúdo e dinamismo do mesmo é criado com recurso a JavaScript e foi utilizado Gradle para gerir as dependências do projeto.

A minha solução conta com:

1. Página inicial que permite visualizar a Homepage, anúncios, registar utilizadores e fazer log-in, não sendo necessária a autenticação do utilizador para utilizar estas ferramentas, exceptuando claro, aquando do log-in.
2. Área de utilizador registado comum que permite além das funcionalidades de utilizador não registado, criar anúncios e enviar/consultar mensagens para anúncios. Para aceder a esta área basta ao utilizador ter credenciais de acesso, que podem ser criadas através da página de registo.
3. Área de administrador que permite além de todas as funcionalidades mencionadas anteriormente, ativar e desativar anúncios. Para ter acesso a esta área só mediante autenticação com credenciais especiais que foram criadas por mim. (Neste caso admin/admin).

O trabalho foi maioritariamente baseado na atividade 17 das aulas práticas.

Classes

- SpringSecurityFormBasedJdbcUserDetailsServiceAuthApp – Classe para iniciar a aplicação web.
- SecurityConfig – Classe que contem a política de segurança da aplicação web e gerenciamento de autorizações para cada perfil.
- CustomErrorController – Classe responsável por mapear os tipos de erros com as páginas a mostrar.
- SpringSecurityController – Classe com o controlador da aplicação web, é responsável por tratar dos pedidos oriundos da vista e tratamento dos mesmos.
- AnunOp – Classe com as operações a realizar sobre as tabelas de anúncios e mensagens na base de dados.
- UserDao – Classe com as operações sobre a tabela de utilizadores da base de dados.
- Anuncio – Classe que representa um anuncio.
- User – Classe que representa um utilizador.
- UserRowMapper – Mapeamento do utilizador quando retornado da bd.
- UserAuthService – Classe responsável por autenticar o utilizador.

JavaServer Pages

- user_interface – Página de utilizador registado
- rgs_msg- Trata do envio de mensagens
- register – Trata de novos registos
- regist_anun – Trata de novos anúncios
- own_msgs – Vista de mensagens do utilizadores
- newuser – Permite o registo de novos utilizadores
- login – Permite o login
- index – Pagina de utilizador não registado
- gestao_anuncios – Pagina para gestão de anúncios ativos e inativos
- error – Tratamento de erros
- criar_anuncio – Permite criar novos anúncios
- anuncios – Pagina para procurar anúncios sendo utilizador não registado
- anuncios_user – Pagina para procurar anúncios sendo um utilizador registado.
- anuncios_sub/anuncios_sub_user – Tratamento de procura de anúncios de utilizador não registado/registado.
- anuncio_alterado – Tratamento de alterações no estado de anúncios

Base de dados e tabelas

Foram criadas três 4 tabelas no desenvolver deste trabalho, “my_user” responsável por guardar os utilizadores e dados do mesmo, “my_user_role” responsável por guardar o tipo de utilizador, “anuncios” que guarda os anuncios e os respetivos dados de cada e mensagens que guarda as mensagens enviadas para um anuncio. As mesmas podem ser encontradas no script de inicialização dbInitScript.sql .

Segurança

De forma a garantir a máxima segurança possível tomei as seguintes opções:

- Sempre que é realizado o envio de informação do lado do cliente para o lado do servidor é utilizado o método post.
- A palavra passe é encriptada e só depois adicionada à bd.
- Para criar mais administradores só o pode fazer a pessoa com acesso ao servidor.
- São usados csrf tokens.

Conclusão

Após concluir este ultimo trabalho de TW percebi o quanto aprendi nesta cadeira, desde aspetos mais básicos como html e css, até à utilização avançada de JS, frameworks e aspectos de segurança. A meu ver este trabalho está muito bem conseguido e foi implementado tudo o que foi pedido pelo professor no enunciado. Considero que se tivesse de refazer o trabalho do inicio após tudo o que aprendi ao desenvolver este, provavelmente teria optado por diferentes abordagens tais como maior utilização de pedidos ajax e utilização de páginas html ao invés de jsp's, desta forma seria mais fácil adaptar partes do primeiro trabalho.