Participantes

amanda caruso cavallaro

samuel pavlovic

****

ATIVIDADE AUTO INSTRUCIONAL DE DESENVOLVIMENTO EM AMBIENTE WEB

e-commerce

BELO HORIZONTE

UNIVERSIDADE FUMEC

2016

Participantes

amanda caruso cavallaro

samuel pavlovic

ATIVIDADE AUTO INSTRUCIONAL DE DESENVOLVIMENTO EM AMBIENTE WEB

e-commerce

Trabalho apresentado à Disciplina de Desenvolvimento em Ambiente Web do Curso de Ciência da Computação, da Universidade FUMEC – FACE.

Período: 5º NA

Professor: Ênio Costa

BELO HORIZONTE  
2016

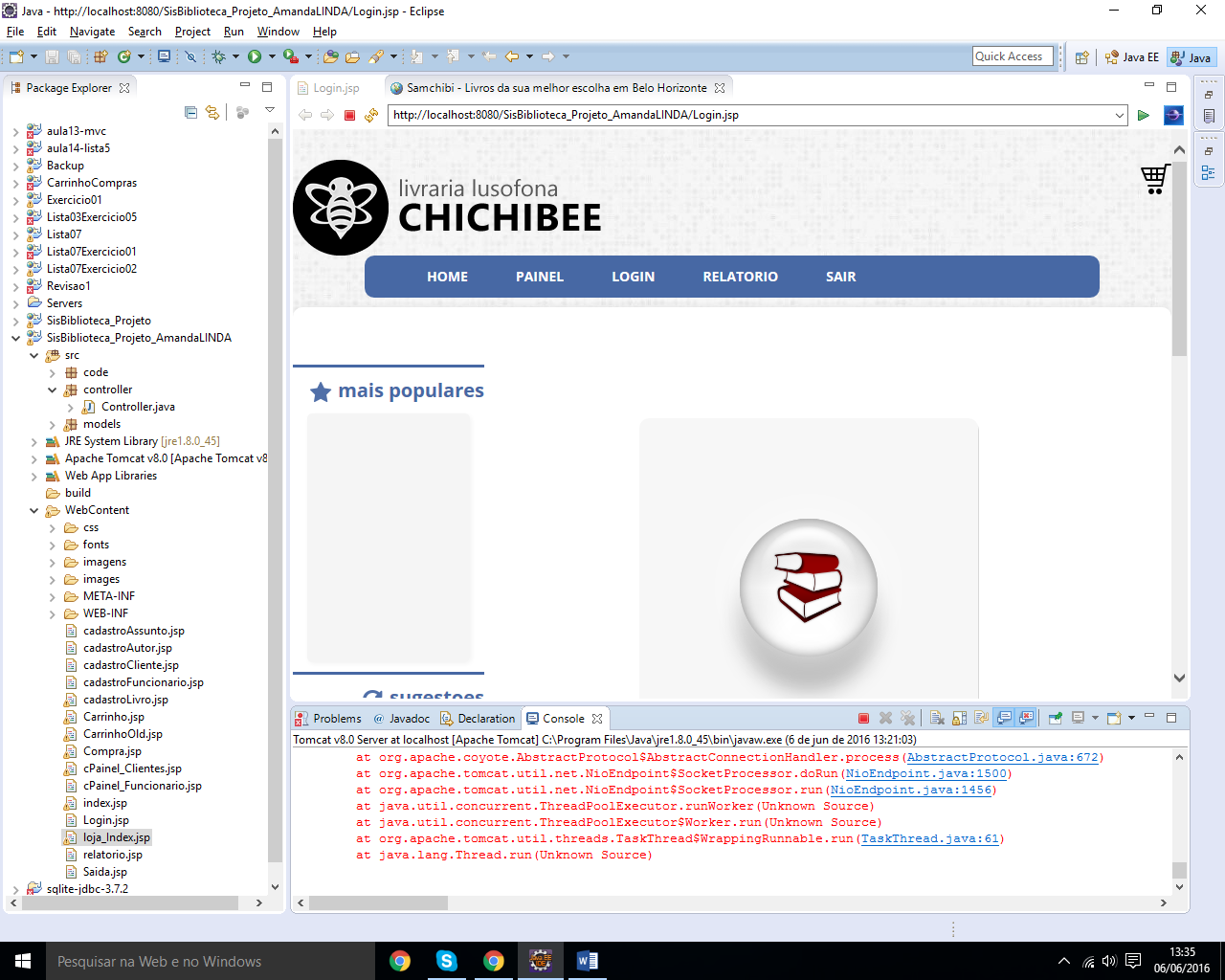
O *e-commerce*, normalmente lida com um grande número de pequenos pedidos, uma grande abrangência geográfica e um consumidor que busca um nível elevado de serviços sem precisar sair do conforto do local que se encontra, a fidelidade passa a ser através do mouse.  As empresas enfrentam dificuldades devido à falta de experiência nesta nova forma emergente de comércio. A adoção de uma estratégia fundamentada na adaptação do varejo tradicional ao varejo virtual é uma ótima forma na qual a nossa página livraria lusofona Chichibee*,* assim como o *Amazon*, *Ebay,* entre outros, encontraram de adaptarem-se às necessidades dos consumidores do século XXI.

O nosso e-commerce consiste na venda de livros on-line, onde utilizamos os seguintes:

Para o Back End utilizamos TomCat com o Eclipse, linguagem de programação Java, seguindo o estilo MVC, com diversas páginas para o Model, View e Controller.

Para a conexão de banco de dados utilizamos o arquivo SISBIBLIOTECA.db e o SqlLiteServer.

Já para o Front End utilizamos jquery, javascript, tags do TomCat para padronizar o design, como também o photoshop para criar a nossa logo e as propagandas laterais.



Abaixo temos a estrutura do nosso projeto dinâmico no Eclipse:

|  |  |  |  |
| --- | --- | --- | --- |
| Views | Controller | Models | Arquivos jsp |
| AssuntoADO.java | Controller.java | AdicionarItensCarrinho.java | cadastroAssunto.jsp |
| AutorADO.java |  | Assunto.java | cadastroAutor.jsp |
| ClienteADO.java |  | Autor.java | cadastroCliente.jsp |
| Conexao.java |  | Cliente.java | cadastroFuncionario.jsp |
| FuncionarioADO.java |  | Funcionario.java | cadastroLivro.jsp |
| LivroADO.java |  | Livro.java | Carrinho.jsp |
|  |  | Produto.java | CarrinhoOld.jsp |
|  |  |  | Compra.jsp |
|  |  |  | cPainel\_Clientes.jsp |
|  |  |  | cPainel\_Funcionarios.jsp |
|  |  |  | index.jsp |
|  |  |  | Login.jsp |
|  |  |  | Loja\_Index.jsp |
|  |  |  | Relatorio.jsp |
|  |  |  | Saida.jsp |

**Instrução:**

Iniciamos o e-commerce na página home index.jsp

Nela, temos na parte superior: o carrinho de compras, a logo com o nome da nossa livraria, e um nav bar com as opções: home, login, segredo e sair.

Se clicar em home, retorna-se para a mesma página.

Se clicar em segredo, temos um easter egg com uma brincadeira.

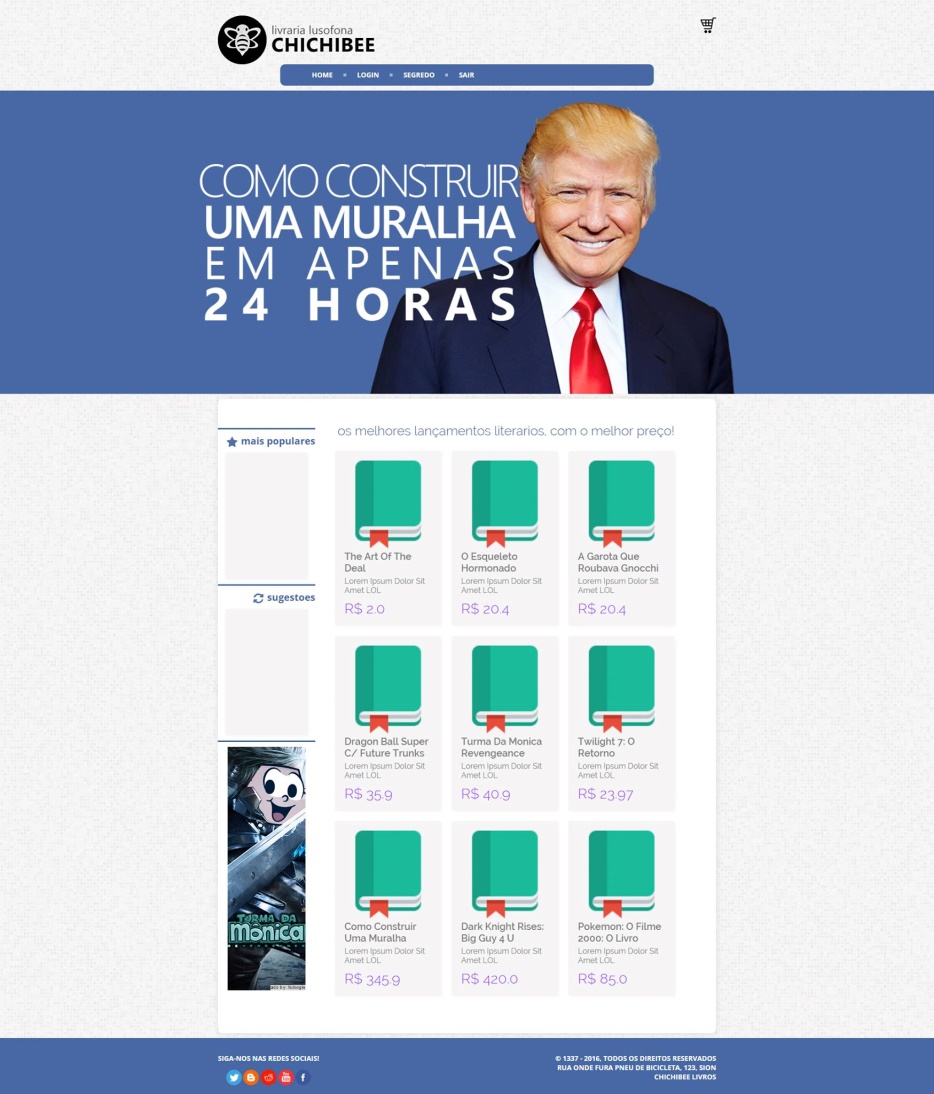
Ao clicar sair é-se desconectado, e redirecionado para home após 5 segundos.



Caso clique em Login, o controller.java o redirecionará para a página Login.jsp

Na parte central temos nossa loja virtual com o acervo dos livros contendo o nome, a descrição e o valor do livro. E nas laterais temos propagandas.

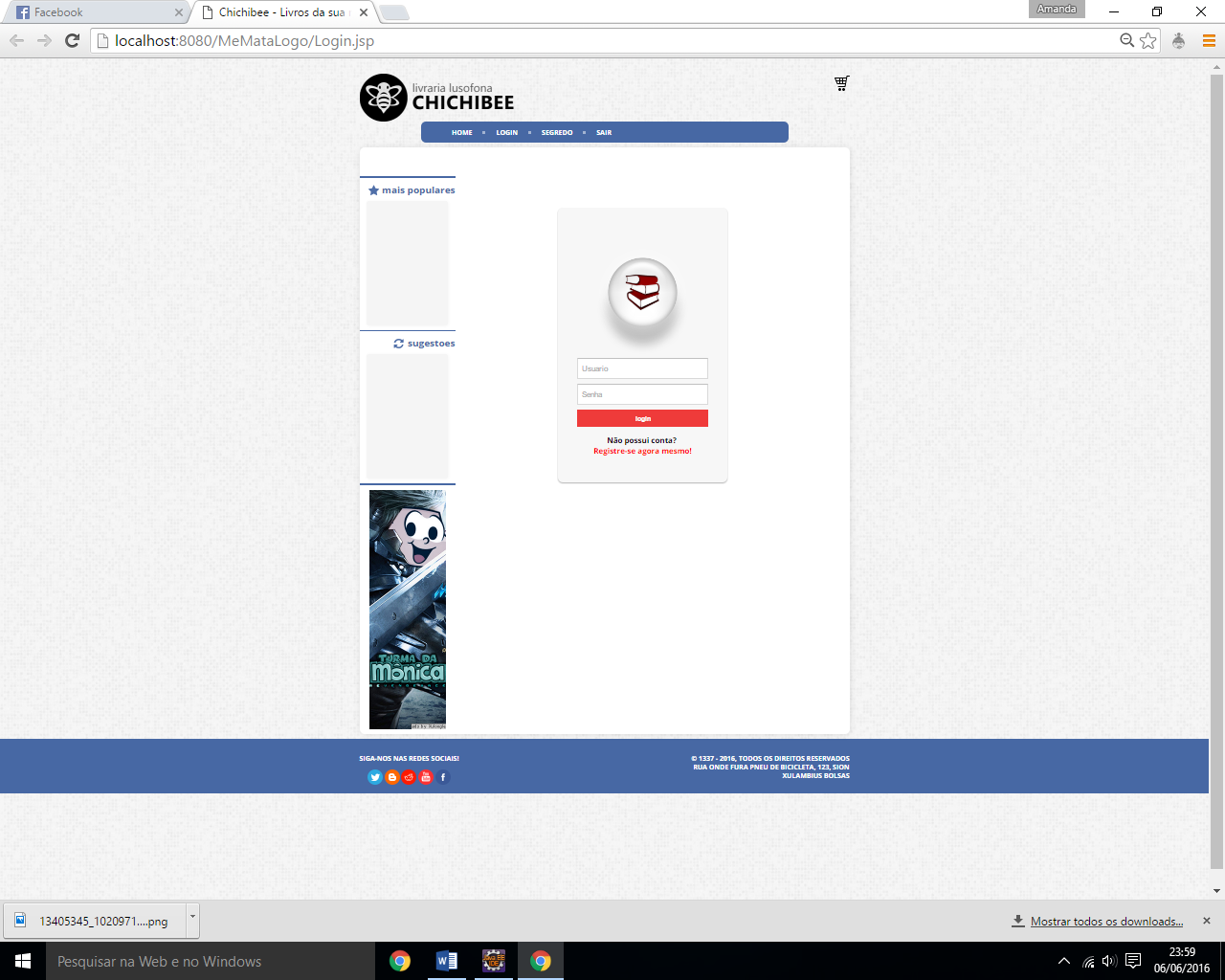
Na parte inferior temos nossos links nas redes sociais e o copyright.



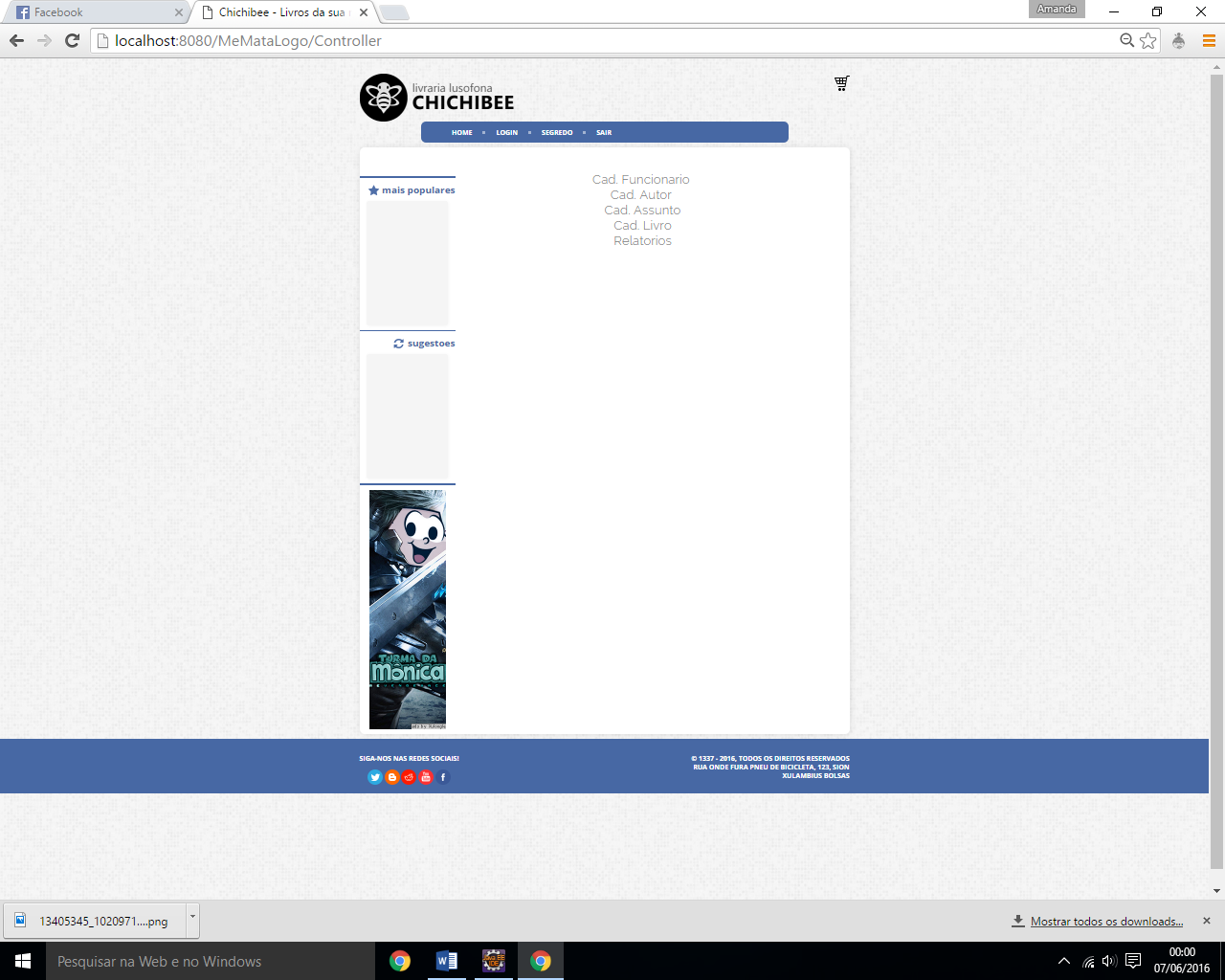
Página Login.jsp

Na página login.jsp pode-se entrar com o login e senha ou criar uma nova conta, caso o login e a senha estejam corretos e você seja um funcionário você será redirecionado para a página em que se pode cadastrar funcionários (cadastroFuncionario.jsp), autor (cadastroAutor.jsp), assunto (cadastroAssunto.jsp), livro (cadastroLivro.jsp) , assim como os relatórios (relatório.jsp) , que é a funcionario.jsp . E ao fazer o cadastro de qualquer um dos anteriormente citados é-se transferido para o index.jsp novamente.

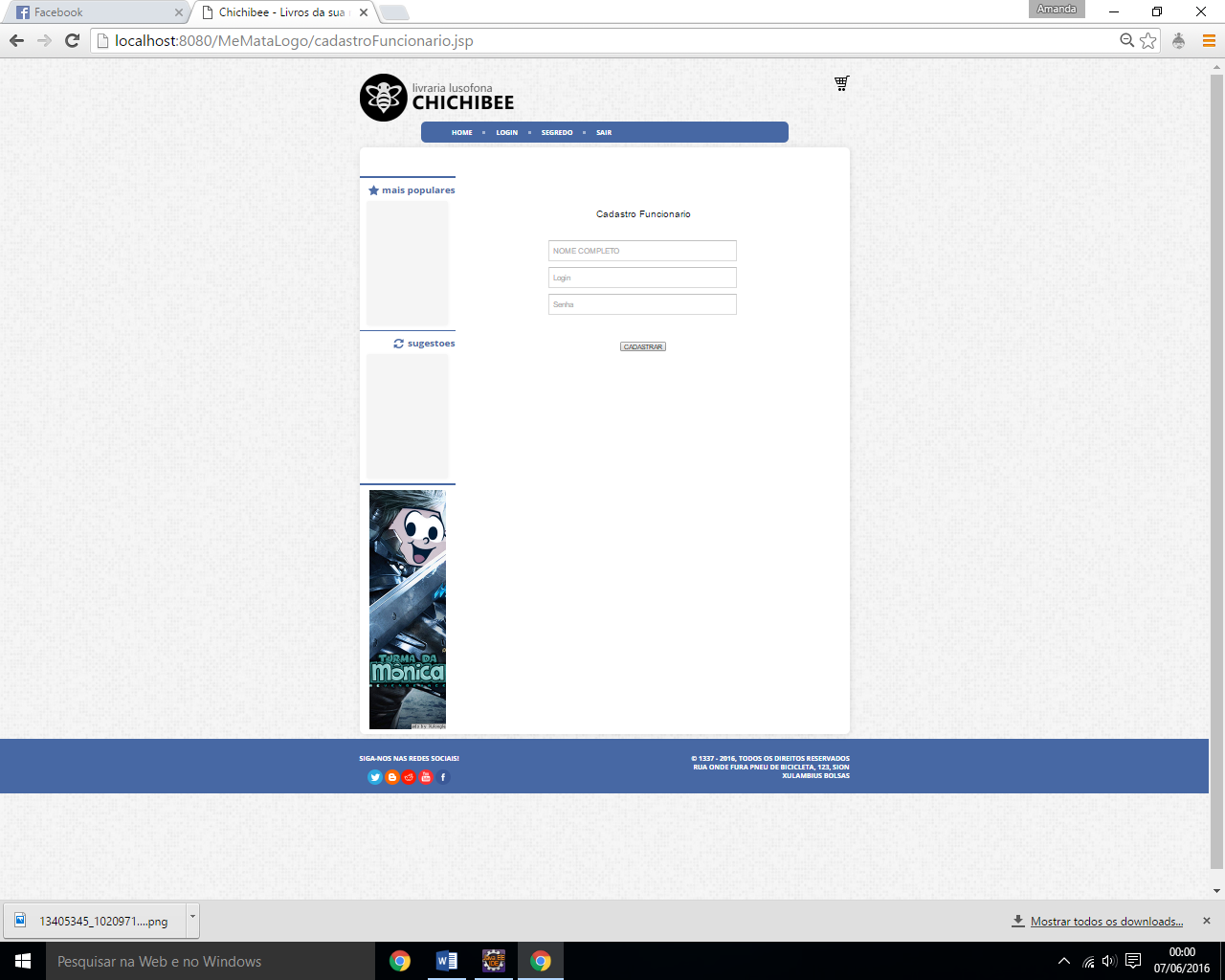
Login.jsp



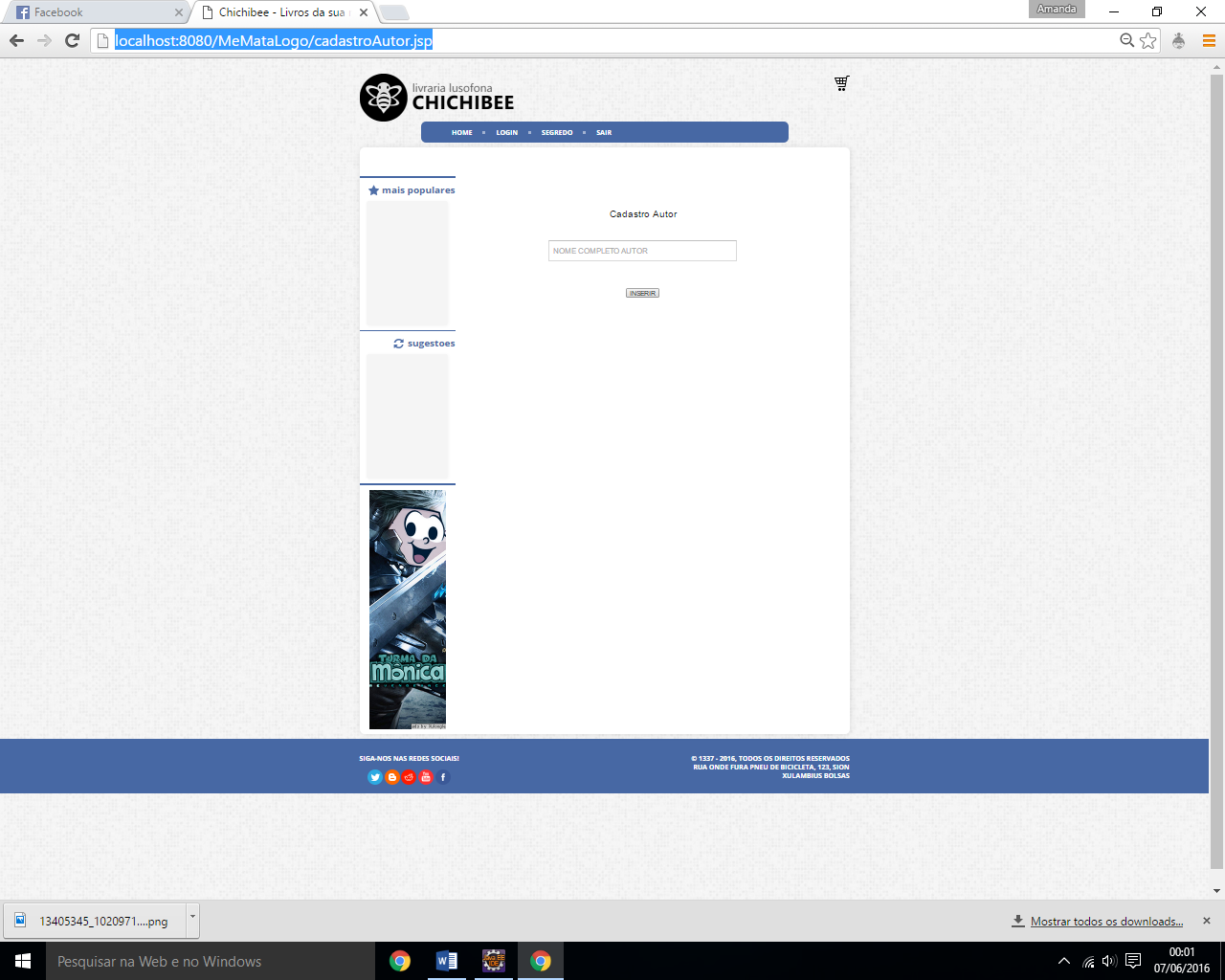
Funcionario.jsp

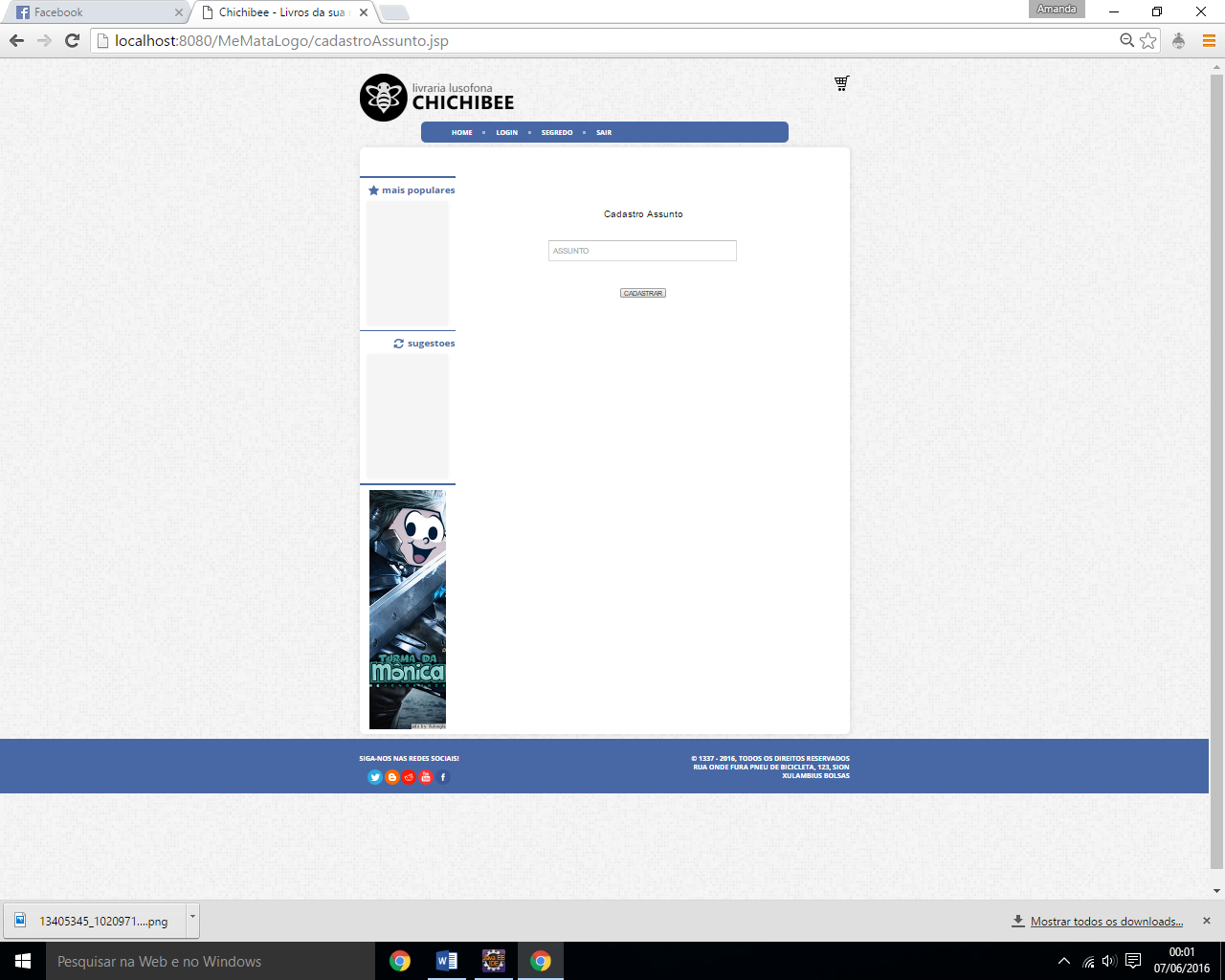


cadastroFuncionario.jsp

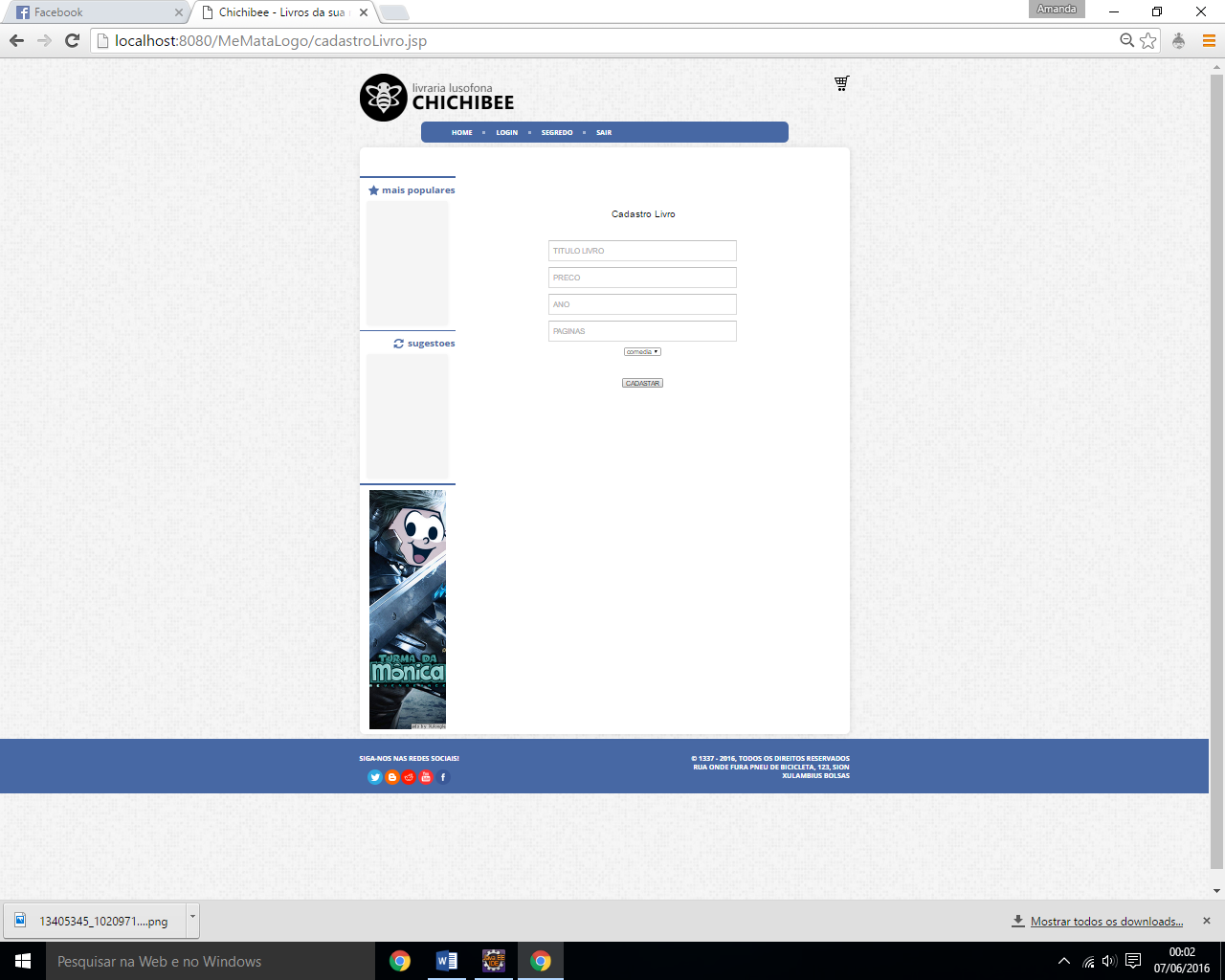


cadastroAutor.jsp

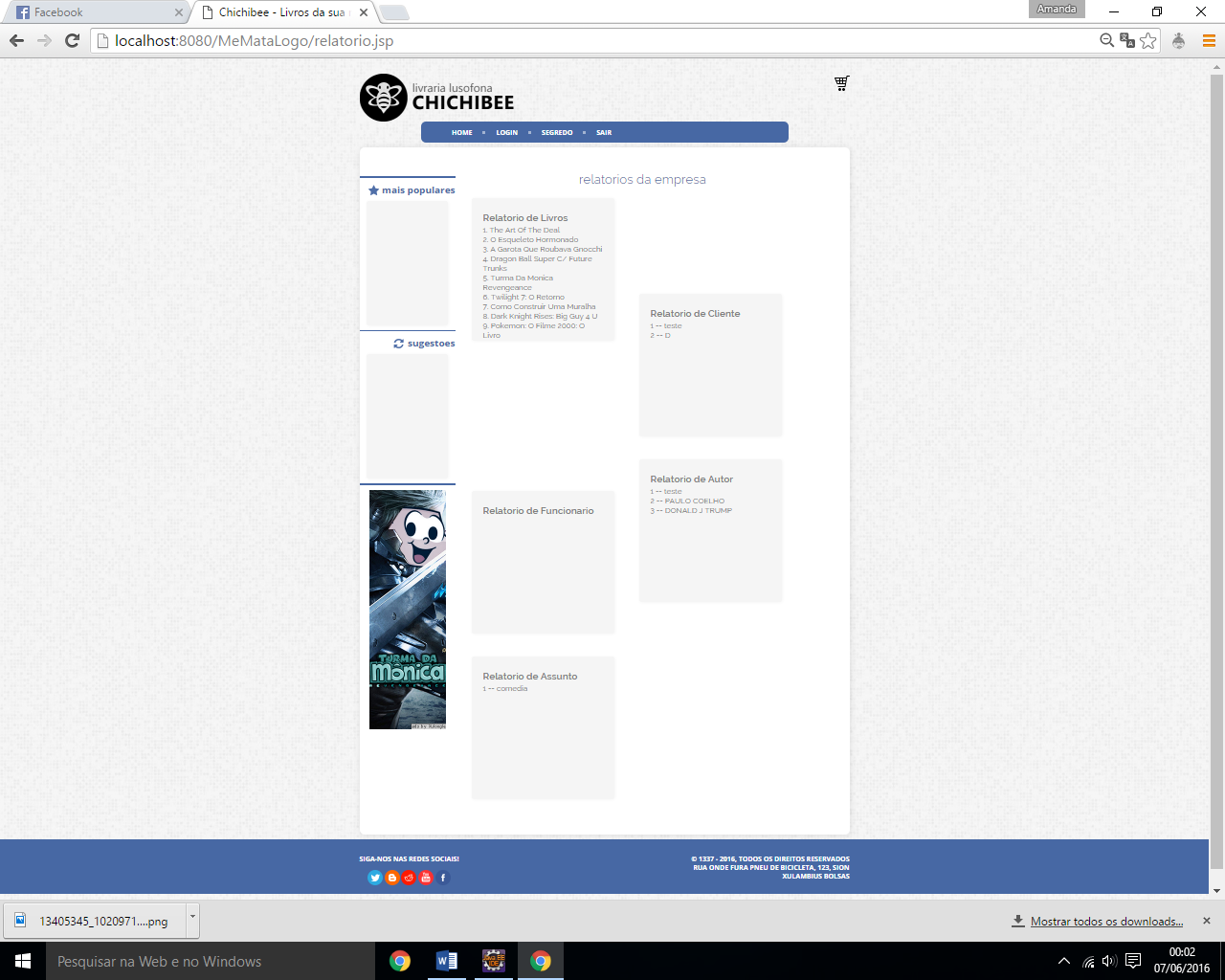


cadastroAssunto.jsp

cadastroLivro.jsp

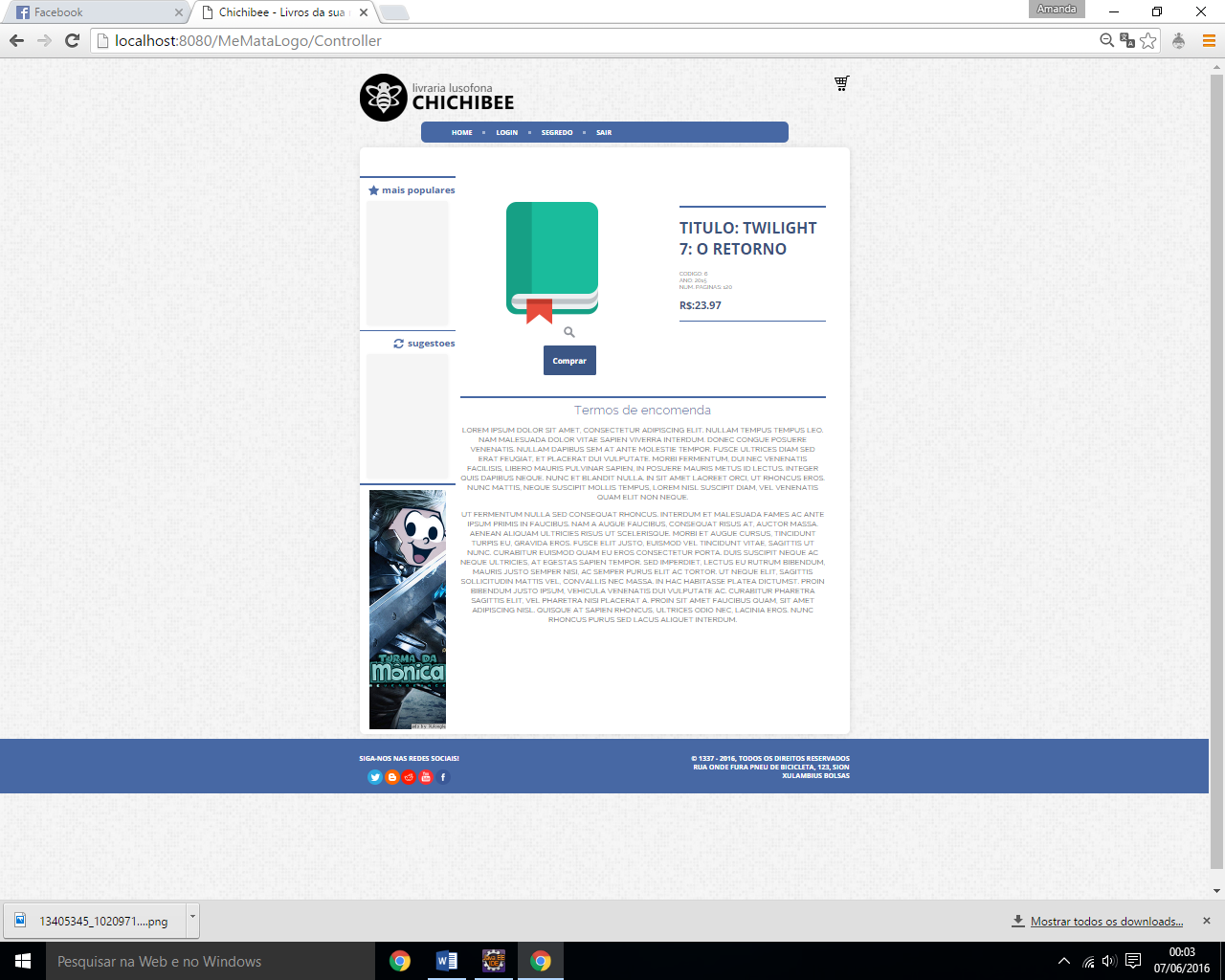


Relatorio.jsp



Ao clicar no livro de escolha, é-se direcionado para a página com maior detalhamento do livro, e ao apertar no botão comprar, é-se transferido para a página de checkout, ao enviar consulta, é-se redirecionado para uma pagina de aviso ao usuario que a compra foi concluida

Compra.jsp



Checkout.jsp



**Descrição breve dos algoritmos e das estruturas de dados utilizadas:**

Nós utilizamos Collection, para suportar operações em coleções de objetos e ArrayList, para implementarmos uma interface de lista. Ambos são usados como forma de guardar e processar os dados referentes aos livros do início ao final da compra.

Utilizamos jquery para facilitar o uso do javascript em nosso site, onde podemos com um código simples fazer operações fortes, especialmente nas páginas index.jsp e loja\_Index.jsp.

**Decisões de implementação não documentadas nesta especificação:**

Nós inicialmente queríamos utilizar Polymer ou bootstrap, porém para melhor nos adaptarmos ao conteúdo e as ferramentas utilizadas em sala de aula, optamos por utilizar apenas as tecnologias que foram vistas em sala, porém aprofundadas com nossos estudos individuais.

**As possíveis melhorias que podem ser feitas são:**

Poderia haver uma consistência mais explícita caso o login esteja incorreto.

Tela para esquecimento de senha

Várias opções de formas de pagamento, com integração com as operadoras de cartão e intermediadoras;

Formas de entrega e frete variadas, com opção de entrega escolhida pelo cliente;

Possuir performance e escalabilidade