



Instituto Superior de Engenharia de Lisboa
Ano Lectivo 2012/2013 – Semestre Inverno

Programação na Internet



Turma: LI51N

Professor:

Luís Falcão

Aluno:

Nuno Assunção – 29199

Índice

Introdução	3
1ª Fase.....	4
2ª Fase.....	5
3ª Fase.....	7
Utilizadores	8

Introdução

Este trabalho pretende emular a aplicação web TRELLO, uma ferramenta para organização de tarefas e objectivos, usando nesta versão final Asp.net MVC para simular a comunicação cliente-servidor. Em seguida irá ser apresentada uma pequena retrospectiva da evolução do trabalho ao longo das duas primeiras fases, aprofundando em seguida as alterações introduzidas na terceira e última fase.

Este trabalho foi desenvolvido tendo por base o trabalho realizado, durante o semestre, pelo grupo 14 (no qual me incluía) da disciplina de Programação na Internet.

1ª Fase

Não sendo a primeira vez que realizávamos a cadeira, decidimos desde o início implementar a organização de classes na forma de MVC para facilitar a integração do trabalho na segunda fase. Foram separadas as diferentes classes em controladores, vistas e modelo, tentando ao máximo manter as funções separadas.

Isto acabou por ser atingido, separando-se ainda dentro da camada de modelo o acesso aos repositórios (Data Access Layer) dos próprios repositórios (Business Layer) através de uma classe que retorna a localização do repositório. Esta fase centra-se na organização e interacção com três entidades: Quadro, Lista e Cartão. Cada lista está contida num quadro e cada cartão está contido numa tarefa.

Em relação ao repositório, podemos salientar a sua estrutura de dados. Foi usado Dicionários para o armazenamento das diferentes entidades, usando os seus nomes como chave. Usando os nomes das entidades em que cada objecto está contido, foram implementados dicionários de listas simplesmente ligadas, no caso das listas, e ainda dicionários de dicionários de listas simplesmente ligadas, no caso das listas e dos cartões, porque era desejado haver uma sensação de sequência de uma entidade dentro da outra.

O uso do nome como chave foi alterado na segunda fase, devido à necessidade de ser possível a edição do campo do nome, algo que isto não permitia, sem acrescentar funções morosas e desnecessárias.

Além das listas que possam ser acrescentadas a um quadro, uma lista é automaticamente adicionada a cada quadro no evento da sua criação: Arquivo. Esta lista irá conter todos os cartões que se desejem arquivar, deixando estes de estar contidos na lista em que se encontravam.

De entre as operações que se podem realizar com estas entidades, destaca-se a possibilidade de mover um cartão para uma localização diferente dentro da própria lista ou até para um índice de outra lista dentro do quadro. É também possível a criação, alteração e eliminação de uma qualquer entidade, dentro de certas circunstâncias.

2ª Fase

Nesta fase, além de se alterar a infra-estrutura usada, de WebGarden para ASP.Net, foram introduzidas mais características ao problema.

Mudámos a chave que é usada para se armazenar as entidades, criando um campo Id nas próprias entidades que é inserido no momento da sua criação.

Foi introduzida a entidade utilizador, que além de ter de conseguir autenticar-se, tem de ter os seus próprios quadros e conseguir partilhá-los com outros utilizadores, especificando o nível de interacção que disponibiliza a esse utilizador.

Foi acrescentado um novo repositório, que irá conter os objectos que representam utilizadores e armazenar também as referências de quadros que são pertinentes para cada utilizador, sejam estes quadros seus ou sejam quadros partilhados por outros utilizadores. Mais uma vez, foram usados dicionários para este fim.

Pelo menos um utilizador terá o grau de administrador, um grau que lhe permite operações adicionais a utilizador, como por exemplo, elevar o grau de um utilizador a administrador.

Teve que se implementar módulos de autenticação e autorização, para ser assegurado que um URI é acedido por alguém com credenciais que lhe permitam o acesso, caso contrário, uma mensagem de erro ou um pedido de autenticação é realizado. Um mesmo URI poderá mostrar vistas diferentes, consoante o grau do cliente que está a realizar o acesso.

Um cliente terá o grau de utilizador apenas depois de se registar no site e passar uma prova enviada para o seu correio electrónico (link de validação).

Criámos um layout diferente, mais parecido com o TRELLO.

Decidimos acrescentar o formulário de edição de quadros, lista e cartões dentro da própria página de detalhes, desde que se tenha permissões para se fazer edição. Estes conseguem detectar se o que está a ser passado nos campos é válido ou não (campo vazio, nome que já exista), mostrando uma mensagem de erro.

Devido à existência de utilizadores e subsequente partilha de quadros, teve que se proceder à criação de entidades DTO (Data Transfer Object),

BoardDetailsDTO, ListDetailsDTO e CardDetailsDTO. Estas entidades conseguem fornecer informações sobre a autoria e autorizações que o utilizador corrente tem sobre a entidade que este DTO encapsula, entre outras. Além disso, Foram criados métodos adicionais no repositório para a inicialização destes DTO não ser feita dentro dos controladores.

Infelizmente, não se conseguiu implementar a imagem dentro do perfil do utilizador.

3ª Fase

Foi implementado código script em JavaScript para se efectuar validações sem se recorrer a pedidos ao servidor. Passando às views informação relevante através de ViewData, validações como perceber se a edição do nome de um quadro é válida já não sobrecarrega desnecessariamente o servidor.

A submissão de formulários (edição de tudo e também a criação de quadros) foi implementada através de pedidos Ajax, não sendo então necessário o recarregamento da página completa. No caso do *browser* não aceitar código em script, o método usado anteriormente mantém-se válido, realizando então um recarregamento completo.

Foi também implementado o mecanismo de Drag-and-drop, usando javascript também, este mecanismo, como tanta coisa, só pode ser usado se o utilizador for o autor do quadro, ou se tiver permissão de edição sobre o mesmo.

Por fim, o mecanismo de pesquisa automática também está a funcionar, usando 3 caracteres, pesquisa automaticamente por quadros, lista e cartões com esse nome.

Estes ficheiros javascript encontram-se na pasta script do projecto.

Utilizadores

A aplicação contém os seguintes utilizadores por defeito:

Nome utilizador: Luís Falcão

Nickname: falcao

Password: slb

Role: admin

Nome utilizador: Nuno Assunção

Nickname: nuno

Password: password

Role: user

Nome utilizador: Vitor

Nickname: vitor

Password: 123456

Role: user

Nome utilizador: Patrick Costa

Nickname: patrick

Password: 123456

Role: admin