

Projeto Prático

Porto, 05 de Abril de 2024

1 OBJETIVO

O objetivo deste projeto é criar uma aplicação web utilizando Java para um blog, com o intuito de proporcionar aos utilizadores uma plataforma interativa para publicar e partilhar conteúdo. O desenvolvimento do projeto será dividido em três partes distintas, permitindo uma abordagem faseada para a implementação gradual das funcionalidades.

1.1 AS DIFERENTES PARTES DO PROJETO

Como referido o projeto está dividido em 3 partes:

1.1.1 Desenvolvimento do Protótipo de Alta-Fidelidade (Parte I):

Nesta etapa, o foco será a criação de um protótipo de alta-fidelidade que apresentará uma proposta visual do layout do blog. Este protótipo irá oferecer uma visão geral das diferentes páginas do blog, incluindo os posts, seus autores, comentários e datas de criação.

O protótipo deverá conter um menu de navegação que permita a transição suave entre as diversas páginas criadas. Quaisquer imagens necessárias para a visualização do layout devem ser carregadas a partir de uma pasta dentro da aplicação.

Para garantir uma apresentação coerente e estilizada, os estilos e formatações das páginas devem ser definidos em um ou mais arquivos CSS, os quais serão incluídos nos arquivos HTML conforme necessário.

Na parte I do projeto, o conteúdo do blog será fictício. Portanto, ele pode ser inserido diretamente no HTML ou injetado no HTML por meio de JavaScript ou através de uma framework de JavaScript. Caso optem pela segunda alternativa, dados fictícios públicos podem ser utilizados, como os disponibilizados pelo site [JSON Placeholder](#), ou então podem ser criados arrays/objetos em JavaScript contendo os dados a serem apresentados.

Assim como nos estilos e formatações, o código JavaScript deve ser organizado em um ou mais arquivos JavaScript e incluído nos arquivos HTML conforme necessário. Qualquer animação ou interação presente no protótipo deve explorar as capacidades do CSS e JavaScript para proporcionar uma experiência dinâmica e cativante aos utilizadores.

1.1.2 Desenvolvimento do Backend e Base de Dados (Parte II):

Nesta fase do projeto, o foco será o desenvolvimento do modelo de dados, a implementação da base de dados, a lógica de negócio e o acesso/modificação dos dados do blog. O backend da aplicação será construído seguindo as melhores práticas e as regras lecionadas durante o curso.

Utilizando a framework SpringBoot, serão criadas APIs REST para facilitar a comunicação entre o frontend e o backend da aplicação. Spring Boot é uma framework que simplifica o desenvolvimento de

aplicações Java, oferecendo uma estrutura robusta e flexível para a criação de serviços web. Poderão ser utilizados os seguintes componentes que julgarem necessários:

- **Spring Data JPA:** Para o acesso aos dados da base de dados de forma simplificada, permitindo a criação de repositórios para cada entidade do modelo de dados.
- **Spring MVC (Model-View-Controller):** Será utilizado para criar os controladores responsáveis por receber as requisições HTTP e direcioná-las para os serviços apropriados.
- **Spring IoC (Inversion of Control) Container:** Será utilizado para gerenciar as dependências entre os diferentes componentes da aplicação.
- **Spring Services:** Serão criados para implementar a lógica de negócio da aplicação, separando-a dos controladores e facilitando a reutilização do código.
- **Spring Boot Starter Web:** Será utilizado para configurar a aplicação como uma aplicação web, incluindo as dependências necessárias para a criação de APIs REST.
- **Spring Boot Starter Data JPA:** Será utilizado para configurar o Spring Data JPA e simplificar o acesso aos dados da base de dados.
- **Spring Boot Starter Validation:** Será utilizado para realizar validações nos dados recebidos nas requisições HTTP.
- **Spring Boot Starter Test:** Será utilizado para escrever testes automatizados para os diferentes componentes da aplicação.

Desenvolver o modelo de dados, a base de dados, a lógica de negócio, o acesso/modificação de dados do blog.

O desenvolvimento do backend da aplicação deverá seguir as regras e as boas práticas lecionadas.

Deverá existir uma interface onde o utilizador possa interagir com os dados (sem nenhuma preocupação com os estilos de apresentação dos mesmos, pois não é o foco desta parte do projeto).

Será necessário ainda a criação de um, ou vários, webservices que devolvam informação sobre os posts, autores, comentários (e outra informação que julguem pertinente).

Modelo e base de dados

Para o modelo de dados, sugere-se a criação dos seguintes modelos, entidades e objetos na base de dados:

- Pessoas
- Utilizadores
- Posts
- Comentários
- Categorias

Adicionalmente, podem ser criados mais modelos, entidades ou objetos na base de dados conforme necessário.

Relacionamentos:

- Cada utilizador corresponde a apenas uma pessoa.
- Um utilizador poderá escrever nenhum, um, ou vários posts.
- Um utilizador poderá escrever nenhum, um, ou vários comentários.

- Um post poderá ter nenhum, um, ou vários comentários.
- Cada post terá apenas um autor.
- Cada está associado a uma ou mais categorias.

O acesso à base de dados será realizado utilizando JPA (Java Persistence API), integrado com o Spring Data JPA para facilitar a interação com o banco de dados de forma eficiente e elegante.

Condicionantes

Nesta fase ainda não é importante interfaces gráficas elaboradas ou animações.

Para segurança do nosso projeto, deverá ser criada uma pasta com o nome “data” onde deverá ser colocado o script SQL que constrói o nosso modelo de dados.

Para facilitar a configuração da aplicação e recursos externos, como por exemplo, obtenção de ligações à base de dados, esta configuração deverá ser feita num ficheiro apropriado para o efeito.

1.1.3 Desenvolvimento do Frontend e Aperfeiçoamento do Backend (Parte III):

Nesta fase do projeto, o objetivo é avançar com o desenvolvimento da parte de frontend da aplicação, utilizando como base o protótipo construído na parte I. Além disso, serão implementadas melhorias no backend, visando garantir a segurança, funcionalidade e encapsulamento da lógica de negócio.

Tarefas no Backend:

- Desenvolvimento de funcionalidades e páginas em falta, que não tenham sido realizadas na parte anterior do projeto.
- Garantir que as boas práticas adotadas anteriormente estão alinhadas com o que foi lecionado, assegurando assim a consistência do código e a manutenibilidade da aplicação.
- Reforçar a segurança do backend, restringindo o acesso por meio de autenticação e perfis de acesso para determinadas funcionalidades.
- Validar se a lógica de negócio está devidamente encapsulada do restante da aplicação. Caso contrário, proceder ao encapsulamento necessário para garantir a coesão e a modularidade do código.
- Assegurar que as operações realizadas pelo Spring Data JPA na base de dados estão a ser executadas dentro de transações (quando necessário), a fim de garantir a consistência dos dados em todos os momentos.

Tarefas no Frontend:

- Garantir que os webservices criados anteriormente estão operacionais e desenvolver os que estiverem em falta.
- Criar a interface da aplicação, baseando-se no protótipo desenvolvido na Parte I do projeto. Para isso, há duas abordagens possíveis:
 - Reformular todas as páginas do protótipo utilizando JSF (JavaServer Faces) e criar novas páginas, se necessário, utilizando a mesma tecnologia.
 - Criar todas as páginas por meio de uma framework de JavaScript, como o React, que interagirá com os webservices já criados e/ou com outros que possam ser pertinentes.

A escolha do React como framework para o desenvolvimento do frontend proporcionará uma experiência de utilizador interativa e responsiva, além de facilitar a integração com os webservices e a criação de interfaces dinâmicas e modernas.

Base de dados (Modelos e Entidades)

Caso seja necessário algum ajuste, este poderá ser efetuado de forma a um melhor desempenho e comportamento.

Estrutura

A escalabilidade e a interoperabilidade do sistema não deverão ser prejudicadas pelas novas funcionalidades e alterações.

Os webservices, preferencialmente deverão ser serviços REST, embora não seja obrigatório usarem esta tecnologia.

A segurança da aplicação deverá/poderá ser feita através da tecnologia que se sentirem mais confortáveis a utilizar.

Condicionantes

Nesta fase é importante a interface gráfica de apresentação e interação do utilizador com a aplicação através do browser. Assim, escolhendo utilizar a tecnologia JSF ou React para criar a interface da aplicação, é importante que a mesma já seja consistente de forma a posteriormente possa ser melhorada com o mínimo de alterações.

1.2 USER STORIES

- Como utilizador, posso ler os posts do blog em ordem cronológica (mais recentes primeiro) para obter uma visão dos pensamentos dos administradores.
- Como utilizador, posso ler os comentários de um post em ordem cronológica (o mais antigo primeiro) para obter informações sobre as opiniões dos utilizadores.
- Como utilizador anónimo, posso me registar com um nome de utilizador e senha para que minha identidade seja estabelecida.
- Como utilizador anónimo, posso fazer login com um nome de utilizador e senha previamente registados para me tornar um utilizador registado.
- Como utilizador registado, posso escrever comentários em posts individuais para partilhar a minha opinião sobre o tópico do post com os administradores e outros utilizadores.
- Como administrador, posso nomear utilizadores registados como administradores para que me possam ajudar na edição e gestão do blog.
- Como administrador, posso criar novos posts no blog para partilhar os meus pensamentos com o mundo. Os artigos têm um cabeçalho e um conteúdo (ambos somente texto).
- Como utilizador registado, posso excluir comentários que fiz para fazer justiça à minha opinião alterada.
- Como administrador, posso excluir comentários para garantir a qualidade do conteúdo do meu blog.
- Como administrador, posso editar posts para corrigir erros.

- Como administrador, posso excluir posts, incluindo os comentários associados, para corrigir erros grosseiros.

Bonus: categorias

- Como administrador, posso organizar posts em categorias para criar ordem entre os meus muitos posts. Um post pode pertencer a várias categorias.
- Como utilizador, posso ler artigos de uma categoria selecionada em ordem cronológica para obter informações sobre os pensamentos dos administradores sobre um determinado tópico.

Bonus: rastrear alterações

- Como utilizador, posso acompanhar as alterações nos posts para verificar a credibilidade dos administradores.

Bonus: favoritos

- Como utilizador, posso adicionar posts a uma lista pessoal de favoritos. Esta lista é mostrada para mim, classificada de acordo com a hora em que foi adicionada à lista.
- Como utilizador, posso ver quantos utilizadores adicionaram uma quantia à sua lista de favoritos.
- Como utilizador, posso ver uma lista de posts classificados em ordem decrescente de acordo com o número de favoritos.

Bonus: moderação

- Como administrador, posso nomear utilizadores registados como moderadores para que me possam ajudar a controlar os comentários.
- Como moderador, posso excluir comentários para garantir a qualidade do conteúdo do blog.

Bonus: agendamento de publicações

- Como administrador, posso escrever posts com antecedência e definir a data de publicação para ser possível apresentar novos conteúdos no meu blog durante as minhas férias.

Bonus: SEO URLs

- Para a otimização do mecanismo de busca do blog, os URLs para visualização dos posts devem conter o título do post, tanto quanto possível. Se o título do post for alterado, o URL também deve ser ajustado, mas os links antigos ainda devem funcionar.

Bonus: upload imagem

- Como administrador, posso carregar uma imagem ao criar ou editar um post, que é exibida aos leitores para tornar o meu post visualmente atraente.

2 CONDIÇÕES DE REALIZAÇÃO E PRAZOS

- O projeto deverá ser realizado individualmente.
- Para cada parte do projeto deverá ser entregue (por via a definir) num ficheiro .zip em que o nome do ficheiro será o vosso primeiro e último nome e um pequeno vídeo em que apenas seja mostrada o projeto a funcionar (não é necessário ter som, nem ver o autor do projeto).

- O projeto deve ser entregue até ao fim da última sessão do CRE.

3 VALORIZAÇÕES

Podem ser efetuados alguns elementos de valorização no projeto, a saber:

- Hospedar a base de dados do projeto num servidor externo, e colocar o projeto a aceder à nossa base de dados lá colocada.

Bom trabalho 😊