



FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Projeto - Meta 2

Sistemas Distribuídos
Licenciatura em Engenharia Informática
2022/2023

Ana Rita Martins Oliveira - 2020213684 - anaoliveira@student.dei.uc.pt
Hugo Sobral de Barros - 2020234334 - hugobarros@student.dei.uc.pt



Índice

1. Arquitetura de software	2
1.1. Estrutura de models	2
1.1.1. HackerNewsItemRecord	2
1.1.2. HackerNewsUserRecord	2
1.2. Views	2
1.2.1. Gets	2
1.2.2. Posts	2
1.2.3. Meta 1	2
1.2.4. Meta 2	2
1.3. Controllers	3
1.3.1. GreetingController	3
1.4. Organização do código	3
2. Integração do Spring Boot com o Servidor RMI da primeira meta	3
3. Integração com o serviço REST	3
3.1. Pesquisar por termos	3
3.2. Pesquisar por utilizador	4
4. Testes	4
4.1. Indexar manualmente um novo URL	4
4.2. Indexar recursivamente todos os URLs encontrados	4
4.3. Pesquisar páginas que contenham um conjunto de termos	4
4.4. Resultados de pesquisa ordenados por importância	5
4.5. Consultar lista de páginas com ligação para uma página específica	5
4.6. Página de administração atualizada em tempo real	5
4.7. Indexar URLs das top stories que contenham os termos da pesquisa	5
4.8. Indexar todas as “stories” de um utilizador	5
5. Conclusão	6

1. Arquitetura de software

1.1. Estrutura de models

1.1.1. HackerNewsItemRecord

Neste model guardamos as estatísticas de um item que recebemos da API do Hacker News. Estas estatísticas incluem, por exemplo, o id, a descrição e o título.

1.1.2. HackerNewsUserRecord

Este model guarda o id de um item recebido da API, bem como, por exemplo, uma lista de histórias, as votações e os comentários de um utilizador.

1.2. Views

1.2.1. Gets

Os endpoints GET, apenas devolvem os templates HTML que foram criados para a apresentação de informação.

1.2.2. Posts

Os endpoints POST, dependendo da sua funcionalidade indexam ou não o URL devolvido.

1.2.3. Meta 1

Os endpoints criados para utilizar o backend da meta 1 apenas desenvolvem funcionalidades do Client da meta 1, isto é, indexar um URL, procurar por termos e procurar URLs associados. Estes endpoints são auxiliados por uma ligação RMI com o Search Module da meta 1.

1.2.4. Meta 2

Os endpoints criados para a ligação com o Hacker News tratam os dados recebidos da API e enviam os URLs recebidos para indexar, também com o auxílio de uma ligação RMI com o Search Module.

1.3. Controllers

1.3.1. GreetingController

Este controller gerencia todos os endpoints, comunica com a API do Hacker News e comunica por RMI com o Search Module.

1.4. Organização do código

O código está organizado em 2 packages. Uma que inclui a meta 1, com todas as funcionalidades backend e outra, o servingwebcontent, correspondente à meta 2, que contém o controller, os models e as templates.

2. Integração do Spring Boot com o Servidor RMI da primeira meta

A integração é feita fazendo:

```
SearchModuleInterface searchModule = (SearchModuleInterface) LocateRegistry.getRegistry(5000).lookup("SearchModule");
```

Esta linha de código, faz com que consigamos aceder a todas as funções do servidor RMI da primeira meta e que as possamos usar nos endpoints necessários. Para tal, basta apenas fazer Search Module seguido de um ponto e da função que se quer usar do RMI da primeira meta, por exemplo, searchModule.addURL(key).

3. Integração com o serviço REST

3.1. Pesquisar por termos

Ao ser recebida uma palavra são percorridas todas as histórias existentes, sendo cada uma delas aberta para se proceder à comparação do título com o termo que se está a pesquisar. Se tal acontecer, é enviada uma lista de listas com as informações necessárias.

3.2. Pesquisar por utilizador

Ao enviar um utilizador é nos devolvida uma lista de submissões que é percorrida e apenas é apresentado algo se existir um título ou um URL, pois apenas notícias possuem título.

4. Testes

4.1. Indexar manualmente um novo URL

- **Objetivo:** Um utilizador deve poder introduzir manualmente um URL para ser indexado.
- **Estado:** Aprovado
- **Objetivo:** O URL indexado é visitado por um Downloader (Web crawler) e ficará associado, no índice invertido, às palavras que forem encontradas no texto.
- **Estado:** Aprovado

4.2. Indexar recursivamente todos os URLs encontrados

- **Objetivo:** O indexador automático deve visitar os URLs que forem encontrados em páginas previamente visitadas.
- **Estado:** Aprovado

4.3. Pesquisar páginas que contenham um conjunto de termos

- **Objetivo:** O motor de busca consulta o índice invertido e apresenta a lista de páginas que contêm todos os termos da pesquisa.
- **Estado:** Aprovado
- **Objetivo:** Para cada resultado da pesquisa, deve mostrar o título da página, o URL completo e uma citação curta composta por texto da página.
- **Estado:** Aprovado
- **Objetivo:** Os resultados da pesquisa devem ser paginados de 10 em 10.
- **Estado:** Aprovado

4.4. Resultados de pesquisa ordenados por importância

- **Objetivo:** Os resultados de pesquisa devem ser apresentados por ordem de relevância.
- **Estado:** Aprovado

4.5. Consultar lista de páginas com ligação para uma página específica

- **Objetivo:** É possível saber, para cada página resultante de uma pesquisa, todas as ligações conhecidas que apontem para essa página.
- **Estado:** Aprovado

4.6. Página de administração atualizada em tempo real

- **Objetivo:** Todos os utilizadores têm acesso a uma opção de consulta de informações gerais sobre o sistema.
- **Estado:** Reprovado
- **Objetivo:** A informação da página é alterada quando acontecem alterações. Mostrando o estado do sistema, designadamente a lista de Downloaders e de Barrels ativos.
- **Estado:** Reprovado.
- **Objetivo:** Mostrar as 10 pesquisas mais comuns realizadas pelos utilizadores.
- **Estado:** Reprovado

4.7. Indexar URLs das top stories que contenham os termos da pesquisa

- **Objetivo:** O Utilizador deve poder solicitar a indexação dos URLs das “top stories” do Hacker News que contenham (no texto) os termos da pesquisa efetuada
- **Estado:** Aprovado

4.8. Indexar todas as “stories” de um utilizador

- **Objetivo:** Deverá ser possível pedir ao Googol para indexar os URLs de todas as “stories” de um dado utilizador do Hacker News, especificado através do seu nome de utilizador.
- **Estado:** Aprovado



5. Conclusão

Como o nosso trabalho não contava com a página de administração da meta 1, não nos foi possível implementar os WebSockets, nem fazer a sua integração com o servidor RMI.

Contudo, de modo geral, conseguimos implementar as funcionalidades que nos foram pedidas com sucesso.

Este foi um projeto muito interessante de se implementar pois, para além dos conhecimentos adquiridos sobre Sistemas Distribuídos, também conseguimos aprofundamos conceitos relacionados com tecnologias como RMI, Spring Boot e HTML.