

1 2



9 0

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA



Your best Search Engine

Engenharia Informática

Sistemas Distribuídos

Diogo Mota - 2021238235

Jóni Pereira - 2021223430

Marta Antunes - 2021216180

Arquitetura de Software

Meta 1

Como se pode verificar no esquema da Figura 1, a Gateway é o server e elemento central de todo o sistema, atuando como intermediária entre os pedidos do cliente e os dados e funcionalidades da aplicação.

Assim que o cliente opta por fazer um insert, search ou um status check, esse pedido é processado pela Gateway que irá aceder às restantes funcionalidades, Queue e Barrels, de acordo com as necessidades do pedido realizado.

Ao realizar um pedido de insert, por exemplo, a Gateway vai enviar o URL introduzido pelo cliente, se válido, para a Queue e o primeiro Downloader que esteja free vai pegar nesse URL para o processar, devolvendo à Queue um ArrayList com todos os URLs encontrados, evitando repetições, na página anterior de modo a escalonar o processamento e construindo uma Database mais extensa. No caso do search, a Gateway entrega a palavra pedida aos Barrels para que estes realizem a pesquisa e devolvam a lista de URLs que contém as palavras pedidas à Gateway para que esta decida como fazer a interseção dos resultados, de modo que o cliente tenha acesso aos resultados que precisa. O status é também resolvido através da interação da Gateway com os Barrels ativos.

Meta 2

Para as novas implementações usaram-se as seguintes linguagens: Java, HTML/CSS (Thymeleaf) e JavaScript. O framework Spring Boot foi usado para o desenvolvimento do backend, enquanto Thymeleaf foi usado para a renderização do lado do Web Server no frontend.

Na pesquisa de palavras, foi implementada uma nova função. As palavras para além de serem procuradas nos barrels, também são pesquisadas usando a API do HackerNews. Todos os resultados são exibidos no nosso browser.

O status nesta meta foi alterado. Existe agora uma página que é atualizada em tempo real. Para isto ser possível, estão a ser usadas WebSockets.

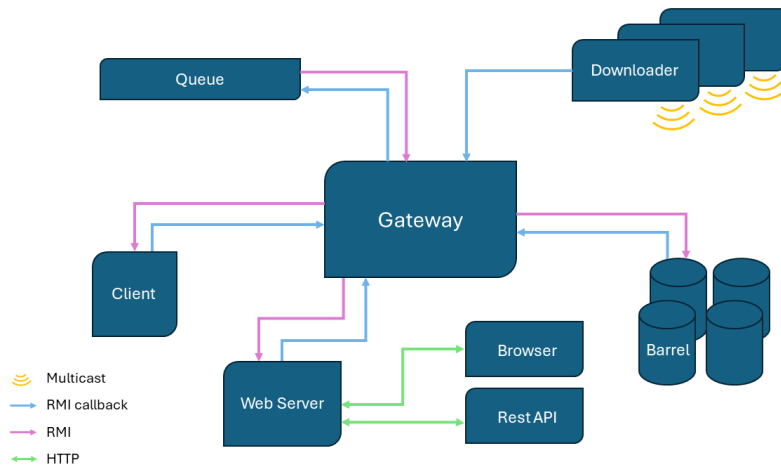


Fig 1 - Esquema da Arquitetura do Sistema implementado

Controllers

[MessagingController.java](#)

Controlador principal que lida com as solicitações HTTP. Tem métodos para inserir URLs, realizar buscas e exibir a página de status.

[HackerNewsController.java](#)

Controlador que lida com as solicitações para a API do HackerNews. Tem métodos para buscar as stories mais recentes para mais tarde serem processadas.

Outros ficheiros importantes

[search.html](#)

Template Thymeleaf para a página de busca. Contém o formulário de busca, a lista de resultados e os botões de navegação.

Serviço REST

Para o HackerNews, o controlador `HackerNewsController` contém um método `hackerNewsTopStories()`. Este método usa a API REST do HackerNews para obter as top

stories. O resultado obtém-se enviando um GET request para o endpoint “/v0/topstories” da API usando a classe RestTemplate do Spring. A resposta é então mapeada para uma lista de HackerNewsItemRecord.

O método hackerNewsTopStories() também suporta a pesquisa de palavras nas top stories da mesma forma que era feita nos barris.

Quanto à API da Wikipedia, apenas um GET request é feito para o necessário, retornando para a página do artigo, com os termos da pesquisa no título, o próprio título e URL juntamente com um pequeno parágrafo de contexto.

Em geral, a API REST é usada neste projeto para obter dados de serviços externos. Esses dados são então usados para fornecer funcionalidades ao usuário.

Testes Realizados

Teste	Descrição	Resultado
1	Ir buscar as stories do Hacker News que contêm todos os termos da pesquisa	Passou
2	Ir buscar o melhor artigo da Wikipedia que contem todos os termos da pesquisa (outra funcionalidade REST)	Passou
3	Aparecerem todas as informações na página do Status antes do cliente fazer uma pesquisa, pe.	Falhou
4	Botão de Next funcionar mais que uma vez	Falhou
5	Integração de uma AI	Falhou
6	Inserir URLs e fazer pesquisas no Web Browser	Passou

Problemas Identificados

O projeto tem um problema com a renderização do template Thymeleaf. O erro ocorre ao tentar analisar a expressão de mudança de páginas no template search.html. Isso é causado porque a variável page é null quando a expressão é avaliada. Uma solução proposta foi garantir que a variável page seja corretamente definida no contexto antes que o template seja renderizado, mas mesmo assim não ficou a 100%.