experts club

Criando Interceptor e Filtro com Spring

24/02/2022

Agenda

experts club

- Sobre mim e a minha relação com o código;
- O que teremos ao término;
- Filter vs. Interceptor;
- Ordem das Camadas;
- Visão geral de uma requisição;
- Code Time;
- Revisão.

Sobre mim e a minha relação com o código

experts club

- **Nome: Gustavo Figueiredo**
- **Formação: Ciência da Computação e Mestrado na área**
- **Tempo trabalhando como dev: 10 anos**
- **Posição / empresa: Engenheiro de Software Sênior / Dock**
- **Redes sociais:**
  - Plataforma da Rocketseat: https://app.rocketseat.com.br/me/gustavo-figueiredo-07013
  - LinkedIn: https://www.linkedin.com/in/gustavo-figueiredo-8602966b/
  - Github: https://github.com/gustavodsf

Sobre a aula e o que será entregue no final

experts club

- **Ao término teremos uma aplicação que é capaz de capturar um requisição, permitindo adicionar mais informação, validar e logar a mesma.**
- **Abaixo temos o repositório com o código criado nesta aula:**
  - https://github.com/rocketseat-experts-club/spring-filter-24-02-2022
- **Ambiente necessário:**
  - Java (JDK >= 11)
  - IDE de sua preferência

Filtro vs Interceptor

experts club

- **Filtro**
  - Os filtros fazem parte do servidor web e não do framework Spring.
  - Podemos usar filtros para manipular e até impedir que solicitações cheguem a qualquer servlet.
  - Spring Security é um ótimo exemplo de uso de filtros para autenticação e autorização
- **HandlerInterceptors fazem parte do framework Spring e ficam entre o DispatcherServlet e nossos Controllers**
  - preHandle() – Executado antes que o manipulador de destino seja chamado
  - postHandle() – Executado após o manipulador de destino, mas antes do DispatcherServlet renderizar a visualização
  - afterCompletion() – Callback após a conclusão do processamento da solicitação e renderização da visualização
  - Pode ser utilizado para adicionar mais informação a uma requisição, dado que, está inserido no contexto do Spring.

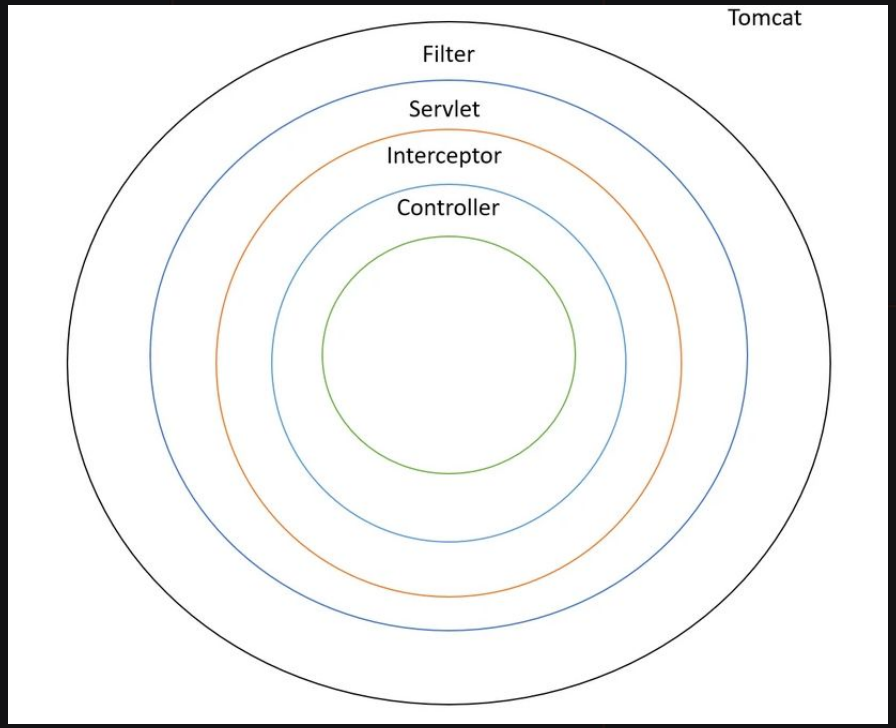
Filtro vs Interceptor

experts club

- **Filtro intercepta as requisições antes que elas cheguem ao DispatcherServlet, tornando-as ideais para tarefas de baixa granularidade, como:**
  - Autenticação
  - Registro e auditoria
  - Compressão de imagem e dados
  - Qualquer funcionalidade que queremos desacoplar do Spring
- **HandlerIntercepors, por outro lado, capta as requisições entre o DispatcherServlet e nossos Controllers.**
  - Sendo realizado dentro estrutura do Spring, fornecendo acesso aos objetos do spring.
  - Permite lidar com preocupações transversais, como registro de aplicativos, envio de mensagens.
  - Verificações de autorização detalhadas
  - Manipulando o contexto ou modelo do Spring

Ordem Das Camadas

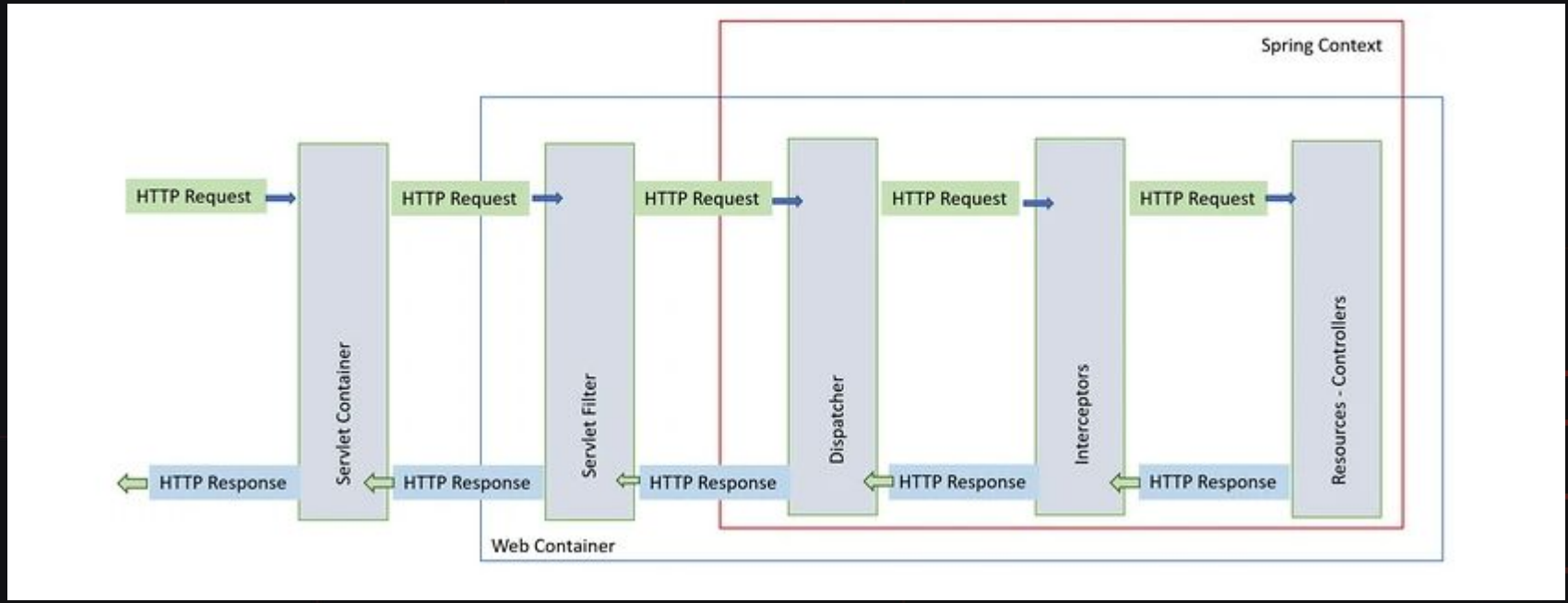
experts club



Fonte: https://segmentfault.com/a/1190000039996829/en

Visão Geral de Uma Requisição

experts club



Fonte: https://segmentfault.com/a/1190000039996829/en

Ordem de Execução dos Métodos

experts club

1. O filtro é chamado pelo contêiner e o método init() do filtro é chamado quando o contêiner é iniciado.
2. O filtro é para todas as solicitações.
3. A ordem de execução de vários filtros é executada na ordem configurada é informada através de uma anotação
4. O interceptor é executado no contexto Spring, quando a requisição é encaminhada ao Spring após o container ser iniciado.
5. A ordem de execução de vários interceptores preHandle é executada na ordem configurada no container.
6. A ordem de execução de vários interceptores afterHandle é realizada na ordem inversa configurada no container.
7. A ordem de execução do afterCompletion de vários interceptores é executada na ordem inversa configurada no container executada após a conclusão de todas as chamadas do afterHandle.

experts club

Code Time!

Revisão

experts club

- Diferença entre Interceptor e Filter.
- Camadas de uma aplicação Spring.
- Visão geral de uma requisição.
- Ordem da Execução do métodos.
- Criação de dois filtros utilizando Spring.
- Criação de um interceptor utilizando Spring.

experts club

Obrigado!

Gustavo Figueiredo
gustavodsf1@gmail.com
Git-Hub: https://github.com/gustavodsf
Linkedin: https://github.com/gustavodsf
https://app.rocketseat.com.br/me/gustavo-figueiredo-07013