Filtres

Les filtres permettent d'intercepter des requêtes HTTP avant qu'elles arrives à leur Servlet afin d'exécuter de la logique. Cela est notamment utile pour restreindre l'accès à certaines routes à des utilisateurs non authentifiés ou ne possèdant pas le bon rôle.

Créer un filtre

Pour créer un filter, il faut créer une classe qui implémente l'interface `Filter` et annotées avec `@WebFilter`. En paramètre de l'annotation, il faut passer le paramètre urlPatterns, pour préciser les routes qui seront filtrée par ce filtre.

```
@WebFilter(urlPatterns = "/admin/*")
public class AdminFilter implements Filter {



    public void doFilter(ServletRequest req, ServletResponse resp, FilterChain chain) throws
ServletException, IOException {

    }

    public void init(FilterConfig config) throws ServletException {

    }

      public void destroy() {
    }

}
```

Par exemple ce filtre va intercepter les requêtes sur toutes les routes qui commencent par "/admin/".

Actions dans un filtre

Dans un filtre on peu récupérer la requête et la réponse HTTP par un simple cast :

```
var request = (HttpServletRequest) req;
var response = (HttpServletResponse) resp;
```

On peut ensuite accèder à la session pour vérfier des informations ou faire une redirection. Pour autoriser la poursuite de la requête normalement utilisez :

```
chain.doFilter();
```

Et pour retourner un code HTTP particulier (401 Unauthorized ou 403 Forbidden) et arrêter là la requête :

```
response.setStatus(HttpServletResponse.SC_UNAUTHORIZED);
return;
```

Revision #3
Created 2020-11-21 23:19:18 UTC by Arsène Lapostolet
Updated 2021-01-16 19:10:22 UTC by Arsène Lapostolet