



Commit

 This commit does not belong to any branch on this repository, and may belong to a fork outside of the repository.

Refactor magic Number

Browse files

 Felipe-Gs committed on Jun 20

1 parent 110d402 commit d665997

Showing 3 changed files with 24 additions and 6 deletions.

Whitespace

Ignore whitespace

Split

Unified

6

openodonto-web/src/main/java/br/ueg/openodonto/controle/busca/SearchableDentista.java

45	45	}
46	46	private static final int NAME_FILTER_MAX_LENGTH = 100;
47	47	private static final int NAME_FILTER_MIN_LENGTH = 3;
	48	+
	49	+
	50	private static final int NAME_FILTER_MAX_LENGTH_ = 150;
	51	+
	52	private static final int NAME_FILTER_MAX_LENGTH__ = 3;
48	52	
49	53	private void buildNameFilter() {
50	54	Validator validator =
		ValidatorFactory.newStrRangeLen(NAME_FILTER_MAX_LENGTH, NAME_FILTER_MIN_LENGTH,
		true);
60	64	}
61	65	
62	66	private void buildEspecialidadeFilter(){
63	-	Validator validator = ValidatorFactory.newStrRangeLen(150,3, true);
	67	+
		Validator validator =
		ValidatorFactory.newStrRangeLen(NAME_FILTER_MAX_LENGTH,NAME_FILTER_MAX_LENGTH_,
		true);
64	68	getFiltersMap().put("especialidadeFilter",
		buildBasicFilter("especialidadeFilter","Especialidade",validator));
65	69	}
66	70	

13

openodonto-web/src/main/java/br/ueg/openodonto/controle/busca/SearchablePessoa.java

```
13      13      import br.ueg.openodonto.persistencia.dao.sql.SqlWhereOperatorType;
14      14      import br.ueg.openodonto.servico.busca.FieldFacade;
15      15      import br.ueg.openodonto.servico.busca.MessageDisplayer;
16      16      + import br.ueg.openodonto.servico.busca.SearchFilterBase;
17      17      import br.ueg.openodonto.validator.ValidatorFactory;
18      18
19      19      public class SearchablePessoa extends AbstractSearchable<Pessoa>{
20      20
21      21          buildCodigoFilter();
22      22
23      23      }
24      24
25      25
26      26      - private void buildNameFilter(){
27      27      -     Validator validator = ValidatorFactory.newStrRangeLen(100,3, true);
28      28      -     getFiltersMap().put("nomeFilter",
29      29      -         buildBasicFilter("nomeFilter","Nome",validator));
30      30
31      31      +     private static final int NAME_FILTER_MAX_LENGTH = 150;
32      32      +     private static final int NAME_FILTER_MIN_LENGTH = 3;
33      33
34      34      +     private void buildNameFilter() {
35      35      +         Validator validator =
36      36      +         ValidatorFactory.newStrRangeLen(NAME_FILTER_MAX_LENGTH, NAME_FILTER_MIN_LENGTH,
37      37      +         true);
38      38
39      39      +         String filterName = "nomeFilter";
40      40      +         String filterDescription = "Nome";
41      41      +         SearchFilterBase nameFilter = buildBasicFilter(filterName,
42      42      +         filterDescription, validator);
43      43      +         getFiltersMap().put(filterName, nameFilter);
44      44
45      45      +     }
46      46
47      47      private void buildCodigoFilter(){
```

11

openodonto-web/src/main/java/br/ueg/openodonto/controle/busca/SearchableProduto.java

```
17      17      import br.ueg.openodonto.persistencia.dao.sql.SqlWhereOperatorType;
18      18      import br.ueg.openodonto.servico.busca.FieldFacade;
19      19      import br.ueg.openodonto.servico.busca.MessageDisplayer;
20      20      + import br.ueg.openodonto.servico.busca.SearchFilterBase;
21      21      import br.ueg.openodonto.validator.ValidatorFactory;
22      22
23      23      public class SearchableProduto extends AbstractSearchable<Produto>{
24      24
25      25          getFiltersMap().put("categoriaFilter",
26      26          buildBasicFilter("categoriaFilter","Categoria",domain,validator));
27      27
28      28      }
29      29
30      30
31      31      +     private static final int NAME_FILTER_MAX_LENGTH = 150;
32      32      +     private static final int NAME_FILTER_MIN_LENGTH = 3;
33      33
34      34      +     private void buildNameFilter() {
```

```
59 - Validator validator = ValidatorFactory.newStrRangeLen(150,3, true);
60 - getFiltersMap().put("nomeFilter",
    buildBasicFilter("nomeFilter","Nome",validator));
63 + Validator validator =
    ValidatorFactory.newStrRangeLen(NAME_FILTER_MAX_LENGTH, NAME_FILTER_MIN_LENGTH,
    true);
64 + String filterName = "nomeFilter";
65 + String filterDescription = "Nome";
66 + SearchFilterBase nameFilter = buildBasicFilter(filterName,
    filterDescription, validator);
67 + getFiltersMap().put(filterName, nameFilter);
61 68 }
62 69
63 70 private void buildColaboradorFilter() {
```

0 comments on commit d665997

Please [sign in](#) to comment.