

E5 - DOCUMENTO DE IMPLEMENTACIÓN

5.1. Introducción

Aquí vamos a incluir el enlace de GitHub con los ficheros del proyecto y los plugins y otras herramientas extra que hemos usado en dicho proyecto.

5.2. Ficheros del proyecto

Todos los ficheros necesarios para el proyecto, tanto los creados gracias al uso del framework *SpringRoo*, como sus comandos, como los ficheros de bases de datos, etc.. se encuentran en la siguiente dirección, dentro del repositorio de GitHub:

<https://github.com/IngenieriaWebUCA/iw2016-facejobs>

A continuación, tenemos que descargarnos los archivos del proyecto con los comandos propios de Git, que son los siguientes:

Para poder descargarse todos los archivos del repositorio el paso a seguir es el de clonar el repositorio a su equipo, mediante el siguiente comando:

git clone <https://github.com/IngenieriaWebUCA/iw2016-facejobs>

5.3. Herramientas del proyecto

Además del uso de alguna distribución GNU-Linux y lo mencionado en el apartado de planificación y configuración, hemos hecho uso de algunas herramientas y plugins para dar funcionalidad extra a nuestro proyecto. Las herramientas y plugins empleados en el proyecto han sido:

- ❖ Spring Security, para restringir el acceso a usuarios autorizados.
- ❖ Spring Roo, como plugin del STS.
- ❖ GitHub, donde se aloja el repositorio de la aplicación.

5.4. La base de nuestro sistema

A continuación, detallaremos la implementación y el desarrollo de nuestro sistema durante la creación de este proyecto. Siguiendo una serie de reglas, realizamos la conexión de nuestro sistema con la base de datos, incluyendo entidades, relaciones y atributos, contando con la seguridad necesaria y desde una aplicación web.

Siendo acorde a los requisitos del proyecto, los desarrolladores de nuestro proyecto describen los pasos realizados de la siguiente manera:

Crear conexión con la Base de Datos

```
jpa setup --provider HIBERNATE --database MYSQL --databaseName facejobs --hostName localhost
```

Creación de la entidad

```
entity jpa --class ~.domain. Entidad --testAutomatically
```

Creación de atributos de las entidades

Ejemplo: definimos un atributo de tipo string, el cuál no puede ser nulo, además de ser único y con un tamaño comprendido de caracteres entre 3 y 64.

```
field string --fieldName nombreAtributo --notNull --unique --sizeMin 3 --sizeMax 64
```

Si nos encontramos fuera de la Entidad sobre la que queremos añadir los atributos correspondientes, únicamente debemos utilizar el siguiente comando para situarnos en la Entidad que necesitamos.

```
focus --class ~.domain. Entidad
```

Definir referencias entre atributos de las entidades

Podemos definir dos tipos.

- ❖ Usando *reference*:

```
field reference --fieldName nombreReferencia --type ~.domain. EntidadReferencia --class ~.domain. Entidad2
```

- ❖ Usando *set*:

```
field set --fieldName formacion --type ~.domain. EntidadReferencia --class ~.domain. Entidad2
```

Generado de capa web

```
web mvc setup
```

```
web mvc all --package ~.web
```

```
web mvc scaffold --class ~.web. EntidadController --backingType ~.domain. Entidad
```

Ámbito de seguridad para posterior configuración

```
roo > security setup
```

Añadido dicho comando, posteriormente definiremos los roles y caparemos el acceso dependiendo del rol de usuario, todo ello se puede hacer mediante la etiqueta <sec: authorize> y modificando el fichero de configuración donde se definen además los roles a utilizar en nuestra aplicación. Dicho fichero será applicationContext-security.xml

5.4.1. Base de datos:

❖ iw2016-facejobs.sql

```
-- phpMyAdmin SQL Dump
-- version 2.10.3
-- http://www.phpmyadmin.net
--
-- Servidor: localhost
-- Tiempo de generación: 05-09-2016 a las 18:21:38
-- Versión del servidor: 5.0.51
-- Versión de PHP: 5.2.6

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

CREATE DATABASE IF NOT EXISTS `facejobs`;

use `facejobs`;

--
-- Base de datos: `facejobs`
--

-----

--
-- Estructura de tabla para la tabla `centro`
--

DROP TABLE IF EXISTS `centro`;
CREATE TABLE IF NOT EXISTS `centro` (
  `id` bigint(20) NOT NULL auto_increment,
  `nombre` varchar(255) NOT NULL,
  `version` int(11) default NULL,
  `id_formacion_demandante_centro` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_sw2cbhqt2k1xb4qw7utqxl0q` (`id_formacion_demandante_centro`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

--
-- Volcar la base de datos para la tabla `centro`
--

-----

--
-- Estructura de tabla para la tabla `demandante`
--

DROP TABLE IF EXISTS `demandante`;
```

```

CREATE TABLE IF NOT EXISTS `demandante` (
  `id` bigint(20) NOT NULL auto_increment,
  `direccion` varchar(255) default NULL,
  `email` varchar(255) default NULL,
  `experiencia` varchar(255) default NULL,
  `fec_nac` datetime default NULL,
  `foto` varchar(255) default NULL,
  `nombre` varchar(255) NOT NULL,
  `resumen_tray` varchar(255) default NULL,
  `telefono` varchar(255) default NULL,
  `tipo_sexo` int(11) NOT NULL,
  `trabajos_disp` varchar(255) default NULL,
  `version` int(11) default NULL,
  `id_experiencia_demandante` bigint(20) default NULL,
  `id_formacion_demandante_demandante` bigint(20) default NULL,
  `id_inscripcion_demandante` bigint(20) default NULL,
  `id_puesto_demandante` bigint(20) default NULL,
  `idusuario_dem` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_a72igk1bbldb34x70a02rqmn9` (`id_experiencia_demandante`),
  KEY `FK_mgtsoqd3rqd71nsn0g795msmw` (`id_formacion_demandante_demandante`),
  KEY `FK_7n75o69u7qfm711twmrne441e` (`id_inscripcion_demandante`),
  KEY `FK_98iks47009lqf0qtgdb75tx75` (`id_puesto_demandante`),
  KEY `FK_d5mxwjex6u3r6vp6l4wm5f9j7` (`idusuario_dem`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;

--
-- Volcar la base de datos para la tabla `demandante`
--

INSERT INTO `demandante` VALUES (1, 'asdasd11', 'gfhg', 'asd', '2016-09-28 00:00:00', 'asdvas',
'jose', 'sdca', '3e23', 0, 'sdfds', 0, NULL, NULL, NULL, NULL, 2);
INSERT INTO `demandante` VALUES (2, 'jh', 'jh', '', NULL, 'DA', 'gonzalo', 'jh', '54', 0, 'aaaa', 0, 1, NULL,
NULL, 1, 1);

-----

--
-- Estructura de tabla para la tabla `empresa`
--

DROP TABLE IF EXISTS `empresa`;
CREATE TABLE IF NOT EXISTS `empresa` (
  `id` bigint(20) NOT NULL auto_increment,
  `act_prof` varchar(255) default NULL,
  `cif` varchar(255) NOT NULL,
  `email` varchar(255) default NULL,
  `nombre` varchar(255) NOT NULL,
  `num_emp` int(11) default NULL,
  `version` int(11) default NULL,
  `idusuario_emp` bigint(20) default NULL,

```

```
PRIMARY KEY (`id`),
KEY `FK_fyuskyhn410ud6kint7vu7auf` (`idusuario_emp`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

```
--
-- Volcar la base de datos para la tabla `empresa`
--
```

```
INSERT INTO `empresa` VALUES (1, 'ua', 'c54543', 'aa@uca.es', 'uca', 12, 0, 1);
```

```
-----
```

```
--
-- Estructura de tabla para la tabla `experiencia`
--
```

```
DROP TABLE IF EXISTS `experiencia`;
CREATE TABLE IF NOT EXISTS `experiencia` (
  `id` bigint(20) NOT NULL auto_increment,
  `fecfin` datetime default NULL,
  `fecini` datetime default NULL,
  `salario` int(11) default NULL,
  `tareass` varchar(255) default NULL,
  `version` int(11) default NULL,
  `id_demandante` bigint(20) default NULL,
  `id_puesto_experiencia` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_b9ofvsh7pm156bkuyg7mijghw` (`id_demandante`),
  KEY `FK_822itl7ly716pmkclojn4p0d6` (`id_puesto_experiencia`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

```
--
-- Volcar la base de datos para la tabla `experiencia`
--
```

```
INSERT INTO `experiencia` VALUES (1, '2016-09-07 00:00:00', '2016-09-07 00:00:00', 123231,
'12312', 1, 1, 1);
```

```
-----
```

```
--
-- Estructura de tabla para la tabla `formacion`
--
```

```
DROP TABLE IF EXISTS `formacion`;
CREATE TABLE IF NOT EXISTS `formacion` (
  `id` bigint(20) NOT NULL auto_increment,
  `nivel` varchar(255) default NULL,
  `titulacion` varchar(255) NOT NULL,
  `version` int(11) default NULL,
  `id_tipo_formacion` bigint(20) default NULL,
```

```

PRIMARY KEY (`id`),
KEY `FK_24unppnu4x02qab7b4vmhe1tn` (`id_tipo_formacion`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

```

```

--
-- Volcar la base de datos para la tabla `formacion`
--

```

```

INSERT INTO `formacion` VALUES (1, 'eesi', 'ingenieria', 0, 1);

```

```

-----

```

```

--
-- Estructura de tabla para la tabla `formaciondemandante`
--

```

```

DROP TABLE IF EXISTS `formaciondemandante`;
CREATE TABLE IF NOT EXISTS `formaciondemandante` (
  `id` bigint(20) NOT NULL auto_increment,
  `version` int(11) default NULL,
  `id_formacion_demandante_centro` bigint(20) default NULL,
  `id_formacion_demandante_demandante` bigint(20) default NULL,
  `id_formacion_demandante_formacion` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_92qp3j7unrp61c1lkx4bl45qx` (`id_formacion_demandante_centro`),
  KEY `FK_g6fdeewx7k49d951iamrmvt0n` (`id_formacion_demandante_demandante`),
  KEY `FK_dad40uvkajv9wubq93p8fe3yq` (`id_formacion_demandante_formacion`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

```

```

--
-- Volcar la base de datos para la tabla `formaciondemandante`
--

```

```

-----

```

```

--
-- Estructura de tabla para la tabla `inscripcion`
--

```

```

DROP TABLE IF EXISTS `inscripcion`;
CREATE TABLE IF NOT EXISTS `inscripcion` (
  `id` bigint(20) NOT NULL auto_increment,
  `estado` int(11) NOT NULL,
  `version` int(11) default NULL,
  `id_demandante` bigint(20) default NULL,
  `id_oferta` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_ojwhoqftbeynjroux6m2cai3l` (`id_demandante`),
  KEY `FK_ds2ahd6b1aihkkodv9n1ggn1k` (`id_oferta`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

```

```
--
-- Volcar la base de datos para la tabla `inscripcion`
--
```

```
INSERT INTO `inscripcion` VALUES (1, 0, 0, 1, 1);
```

```
-- -----
--
-- Estructura de tabla para la tabla `oferta`
--
```

```
DROP TABLE IF EXISTS `oferta`;
CREATE TABLE IF NOT EXISTS `oferta` (
  `id` bigint(20) NOT NULL auto_increment,
  `estado` int(11) default NULL,
  `fec_fin_oferta` datetime default NULL,
  `fec_ini_oferta` datetime default NULL,
  `fecha_inc` datetime default NULL,
  `info_puesto` varchar(255) NOT NULL,
  `num_vac` int(11) default NULL,
  `perfil_dem` varchar(255) default NULL,
  `salario_bruto` int(11) default NULL,
  `tipo_contrato` int(11) default NULL,
  `titulo` varchar(255) NOT NULL,
  `version` int(11) default NULL,
  `id_formacion` bigint(20) default NULL,
  `id_inscripcion_oferta` bigint(20) default NULL,
  `id_puesto_oferta` bigint(20) default NULL,
  `id_sede_oferta` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_7fskxl1k21dd5yu28st5xa69m` (`id_formacion`),
  KEY `FK_ato0c3c6wowf69grguynjbaal` (`id_inscripcion_oferta`),
  KEY `FK_661ulupjggglqul00j9n2yxg0` (`id_puesto_oferta`),
  KEY `FK_p6xltqa341q4h5kxa7srg6s4j` (`id_sede_oferta`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;
```

```
--
-- Volcar la base de datos para la tabla `oferta`
--
```

```
INSERT INTO `oferta` VALUES (1, 0, '2016-09-20 00:00:00', '2016-09-07 00:00:00', '2016-09-07 00:00:00', 'qwe', 1, 'qsdas', 21, 0, 'oferta1', 0, NULL, NULL, NULL, 1);
INSERT INTO `oferta` VALUES (2, 0, '2016-09-20 00:00:00', '2016-09-13 00:00:00', '2016-09-09 00:00:00', 'af', 1231, 'asdfa', 123, 0, 'ppp', 0, 1, NULL, 1, 1);
```

```
-- -----
--
-- Estructura de tabla para la tabla `puesto`
```

```
--
DROP TABLE IF EXISTS `puesto`;
CREATE TABLE IF NOT EXISTS `puesto` (
  `id` bigint(20) NOT NULL auto_increment,
  `version` int(11) default NULL,
  `id_puesto_demandante` bigint(20) default NULL,
  `id_puesto_experiencia` bigint(20) default NULL,
  `id_puesto_oferta` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_cs5i2avsg0s7j0qmfqkvij7kq` (`id_puesto_demandante`),
  KEY `FK_3y6s9dtnhmun40d7nl8dem4jp` (`id_puesto_experiencia`),
  KEY `FK_nlwkvpf1k01fja0tfi78hne77` (`id_puesto_oferta`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

```
--
-- Volcar la base de datos para la tabla `puesto`
--
```

```
INSERT INTO `puesto` VALUES (1, 0, 1, 1, 1);
```

```
-- -----
```

```
--
-- Estructura de tabla para la tabla `sede`
--
```

```
DROP TABLE IF EXISTS `sede`;
CREATE TABLE IF NOT EXISTS `sede` (
  `id` bigint(20) NOT NULL auto_increment,
  `ciudad` varchar(255) default NULL,
  `nombre` varchar(255) default NULL,
  `version` int(11) default NULL,
  `id_empresa` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_t12ftsosrsc3pbnvjqoe7evll` (`id_empresa`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

```
--
-- Volcar la base de datos para la tabla `sede`
--
```

```
INSERT INTO `sede` VALUES (1, 'protreal', 'ptoreal', 0, 1);
```

```
-- -----
```

```
--
-- Estructura de tabla para la tabla `tipoformacion`
--
```

```
DROP TABLE IF EXISTS `tipoformacion`;
```



```

CREATE TABLE IF NOT EXISTS `tipoformacion` (
  `id` bigint(20) NOT NULL auto_increment,
  `nombre` varchar(255) default NULL,
  `version` int(11) default NULL,
  `id_formacion_tipo_formacion` bigint(20) default NULL,
  PRIMARY KEY (`id`),
  KEY `FK_rapslhpx2ays8jsdpbljbu0rq` (`id_formacion_tipo_formacion`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

--
-- Volcar la base de datos para la tabla `tipoformacion`
--

INSERT INTO `tipoformacion` VALUES (1, 'tipo1', 0, NULL);

-----

--
-- Estructura de tabla para la tabla `usuario`
--

DROP TABLE IF EXISTS `usuario`;
CREATE TABLE IF NOT EXISTS `usuario` (
  `id` bigint(20) NOT NULL auto_increment,
  `id_tipo_usu` int(11) NOT NULL,
  `login` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `version` int(11) default NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;

--
-- Volcar la base de datos para la tabla `usuario`
--

INSERT INTO `usuario` VALUES (1, 0, 'e', 'e', 0);
INSERT INTO `usuario` VALUES (2, 1, 'jose', 'jose', 0);

--
-- Filtros para las tablas descargadas (dump)
--

--
-- Filtros para la tabla `centro`
--

ALTER TABLE `centro`
  ADD CONSTRAINT `FK_sw2cbhqt2k1xb4qw7utqxl0q` FOREIGN KEY
  (`id_formacion_demandante_centro`) REFERENCES `formaciondemandante` (`id`);

--
-- Filtros para la tabla `demandante`

```

```

--
ALTER TABLE `demandante`
  ADD CONSTRAINT `FK_d5mxwjex6u3r6vp6l4wm5f9j7` FOREIGN KEY (`idusuario_dem`) REFERENCES
`usuario` (`id`),
  ADD CONSTRAINT `FK_7n75o69u7qfm711twmrne441e` FOREIGN KEY
(`id_inscripcion_demandante`) REFERENCES `inscripcion` (`id`),
  ADD CONSTRAINT `FK_98iks47009lqf0qtgdb75tx75` FOREIGN KEY (`id_puesto_demandante`)
REFERENCES `puesto` (`id`),
  ADD CONSTRAINT `FK_a72igk1bbldb34x70a02rqmn9` FOREIGN KEY (`id_experiencia_demandante`)
REFERENCES `demandante` (`id`),
  ADD CONSTRAINT `FK_mgtsoqd3rqd71nsn0g795msmw` FOREIGN KEY
(`id_formacion_demandante_demandante`) REFERENCES `formaciondemandante` (`id`);

--
-- Filtros para la tabla `empresa`
--
ALTER TABLE `empresa`
  ADD CONSTRAINT `FK_fyuskyhn410ud6kint7vu7auf` FOREIGN KEY (`idusuario_emp`) REFERENCES
`usuario` (`id`);

--
-- Filtros para la tabla `experiencia`
--
ALTER TABLE `experiencia`
  ADD CONSTRAINT `FK_822itl7ly716pmkclojn4p0d6` FOREIGN KEY (`id_puesto_experiencia`)
REFERENCES `puesto` (`id`),
  ADD CONSTRAINT `FK_b9ofvsh7pm156bkuyg7mijghw` FOREIGN KEY (`id_demandante`)
REFERENCES `demandante` (`id`);

--
-- Filtros para la tabla `formacion`
--
ALTER TABLE `formacion`
  ADD CONSTRAINT `FK_24unppnu4x02qab7b4vmhe1tn` FOREIGN KEY (`id_tipo_formacion`)
REFERENCES `tipoformacion` (`id`);

--
-- Filtros para la tabla `formaciondemandante`
--
ALTER TABLE `formaciondemandante`
  ADD CONSTRAINT `FK_dad40uvkajv9wubq93p8fe3yq` FOREIGN KEY
(`id_formacion_demandante_formacion`) REFERENCES `formacion` (`id`),
  ADD CONSTRAINT `FK_92qp3j7unrp61c1lkx4bl45qx` FOREIGN KEY
(`id_formacion_demandante_centro`) REFERENCES `centro` (`id`),
  ADD CONSTRAINT `FK_g6fdeewx7k49d951iamrmvt0n` FOREIGN KEY
(`id_formacion_demandante_demandante`) REFERENCES `demandante` (`id`);

--
-- Filtros para la tabla `inscripcion`
--
ALTER TABLE `inscripcion`

```

```

ADD CONSTRAINT `FK_ds2ahd6b1aihkkodv9n1gggn1k` FOREIGN KEY (`id_oferta`) REFERENCES
`oferta` (`id`),
ADD CONSTRAINT `FK_ojwhoqftbeynjroux6m2cai3l` FOREIGN KEY (`id_demandante`) REFERENCES
`demandante` (`id`);

--
-- Filtros para la tabla `oferta`
--
ALTER TABLE `oferta`
ADD CONSTRAINT `FK_p6xltqa341q4h5kxa7srg6s4j` FOREIGN KEY (`id_sede_oferta`) REFERENCES
`sede` (`id`),
ADD CONSTRAINT `FK_661ulupjggglqul00j9n2yxg0` FOREIGN KEY (`id_puesto_oferta`) REFERENCES
`puesto` (`id`),
ADD CONSTRAINT `FK_7fskxl1k21dd5yu28st5xa69m` FOREIGN KEY (`id_formacion`) REFERENCES
`formacion` (`id`),
ADD CONSTRAINT `FK_ato0c3c6wowf69grguynjbaal` FOREIGN KEY (`id_inscripcion_oferta`)
REFERENCES `inscripcion` (`id`);

--
-- Filtros para la tabla `puesto`
--
ALTER TABLE `puesto`
ADD CONSTRAINT `FK_nlwkvpf1k01fja0tfi78hne77` FOREIGN KEY (`id_puesto_oferta`) REFERENCES
`oferta` (`id`),
ADD CONSTRAINT `FK_3y6s9dtnhmun40d7nl8dem4jp` FOREIGN KEY (`id_puesto_experiencia`)
REFERENCES `experiencia` (`id`),
ADD CONSTRAINT `FK_cs5i2avsg0s7j0qmfqkvij7kq` FOREIGN KEY (`id_puesto_demandante`)
REFERENCES `demandante` (`id`);

--
-- Filtros para la tabla `sede`
--
ALTER TABLE `sede`
ADD CONSTRAINT `FK_t12ftsosrsc3pbnvjqoe7evll` FOREIGN KEY (`id_empresa`) REFERENCES
`empresa` (`id`);

--
-- Filtros para la tabla `tipoformacion`
--
ALTER TABLE `tipoformacion`
ADD CONSTRAINT `FK_rapslhpx2ays8jsdpbljbu0rq` FOREIGN KEY (`id_formacion_tipo_formacion`)
REFERENCES `formacion` (`id`);

```

5.4.2. Restricción por vista de botones de acción (Ejemplo) :

```

<c:if test="{delete}">
    <td class="utilbox">
        <spring:url value="{path}/{itemId}" var="delete_form_url" />
        <spring:url value="/resources/images/delete.png" var="delete_image_url" />
        <form:form action="{delete_form_url}" method="DELETE">
            <spring:message arguments="{typeName}" code="entity_delete" var="delete_label"
htmlEscape="false" />

```

```

<c:set var="delete_confirm_msg">
  <spring:escapeBody javaScriptEscape="true">
    <spring:message code="entity_delete_confirm" />
  </spring:escapeBody>
</c:set>
<input alt="{fn:escapeXml(delete_label)}" class="image" src="{delete_image_url}"
title="{fn:escapeXml(delete_label)}" type="image" value="{fn:escapeXml(delete_label)}"
onclick="return confirm('{delete_confirm_msg}');" />
<c:if test="{not empty param.page}">
  <input name="page" type="hidden" value="1" />
</c:if>
<c:if test="{not empty param.size}">
  <input name="size" type="hidden" value="{fn:escapeXml(param.size)}" />
</c:if>
</form:form>
</td>
</c:if>

```

Generador de buscadores por criterios concretos (finders)

Debemos ejecutar la orden siguiente en la consola de Roo:

finder list

Lo cual nos devolverá un listado de finders o buscadores basados en los atributos de la Entidad en la que estemos situados. Por defecto Roo devuelve finders de profundidad uno, no devolverá finders por criterios múltiples del tipo `findBySexoLikeAndSueldoGreaterThan()`

sino finders del tipo `findBySueldoGreaterThan`, `findBySexoLike`, `findByNombreLike...etc.`

Observamos la lista que nos devuelve y bajo nuestro criterio, elegiremos uno de ellos, lo cuál se le debe indicar para que lo añada mediante la siguiente orden:

finder add nombreDelFinder

En nuestro caso, realizaremos finders bajo múltiples criterios, lo cual quedará modificado bajo nuestra responsabilidad y no la de SpringRoo, ya que no maneja buscadores por más de un criterio.

5.4.5. Idiomas (i18n)

Se han añadido soporte para el idioma inglés, lo cual ha conllevado realizar una traducción del fichero en español al otro idioma.

5.4.6. Fichero de registro de comandos utilizados (log.roo):

```
// Spring Roo
1.3.2.RELEASE
[rev 8387857]
log opened at
2016-08-20
12:40:30
```

```
project --topLevelPackage es.uca.iw --projectName iw2016-facejobs --java
6 --packaging WAR
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-20
12:40:31
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-20
12:40:35
```

//CONEXIÓN A LA BASE DE DATOS

```
jpa setup --provider HIBERNATE --database MYSQL
```

//CREACIÓN DE ENTIDADES Y ATRIBUTOS

```
// [failed] entity jpa --class ~.domain.centro
```

```
// [failed] field number --fieldName id --type java.lang.Integer --
notNull --class ~.domain.centro
```

```
// [failed] entity jpa --class ~.domain.centro
```

```
// [failed] field number --fieldName id --type java.lang.Integer --
notNull --class ~.domain.Centro
```

```
// [failed] field number --fieldName id --type java.lang.Integer --
notNull --class ~.domain.centro
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-20
12:53:43
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-20
12:54:06
```

```
jpa setup --provider HIBERNATE --database MYSQL
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-20
12:54:44
```

```
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-20
12:55:15
```

```
entity jpa --class ~.domain.centro
```

```
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.centro
```

```
field string --fieldName nombre --class ~.domain.centro
```

//CREACIÓN DE BUSCADORES

```
finder add --finderName findId --class ~.domain.centro
```

```
finder add --finderName findIdLike --class ~.domain.centro
```

//INSTALACIÓN DEL MODELO VISTA-CONTROLADOR

```
web mvc setup
```

```
web mvc all --package ~.web
```

```
entity jpa --class ~.domain.empresa
```

```

field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.empresa
field string --fieldName nombre --class ~.domain.empresa
field string --fieldName cif --class ~.domain.empresa
field string --fieldName email --class ~.domain.empresa
// [failed] field string --fieldName email --class ~.domain.empresa
field string --fieldName actProf --class ~.domain.empresa
field number --fieldName numEmp --type java.lang.Integer --class
~.domain.empresa
field number --fieldName idUsuario --type java.lang.Integer --class
~.domain.empresa
web mvc all --package ~.web
entity jpa --class ~.domain.usuario
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.usuario
field string --fieldName login --class ~.domain.usuario
field string --fieldName password --class ~.domain.usuario
field number --fieldName tipo --type java.lang.Integer --class
~.domain.usuario
web mvc all --package ~.web
entity jpa --class ~.domain.demandante
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.demandante
field string --fieldName nombre --class ~.domain.demandante
field date --fieldName facNac --type java.util.Date --class
~.domain.demandante
field number --fieldName sexo --type java.lang.Integer --class
~.domain.demandante
field string --fieldName direccion --class ~.domain.demandante
field string --fieldName email --class ~.domain.demandante
field string --fieldName telefono --class ~.domain.demandante
field number --fieldName idusuario --type java.lang.Integer --class
~.domain.demandante
field string --fieldName resumenTray --class ~.domain.demandante
field string --fieldName foto --class ~.domain.demandante
field string --fieldName trabajosDisp --class ~.domain.demandante
field string --fieldName experiencia --class ~.domain.demandante
web mvc all --package ~.web
entity jpa --class ~.domain.inscripcion
field number --fieldName idcurriculum --type java.lang.Integer --notNull
--class ~.domain.inscripcion
field number --fieldName idoferta --type java.lang.Integer --notNull --
class ~.domain.inscripcion
field number --fieldName estado --type java.lang.Integer --notNull --
class ~.domain.inscripcion

```

```

web mvc all --package ~.web
entity jpa --class ~.domain.sede
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.sede
field string --fieldName nombre --class ~.domain.sede
field number --fieldName idempresa --type java.lang.Integer --notNull --
class ~.domain.sede
field string --fieldName ciudad --class ~.domain.sede
web mvc all --package ~.web
entity jpa --class ~.domain.puesto
field number --fieldName iddemandante --type java.lang.Integer --notNull
--class ~.domain.puesto
field number --fieldName idexperiencia --type java.lang.Integer --
notNull --class ~.domain.puesto
field number --fieldName idoferta --type java.lang.Integer --notNull --
class ~.domain.puesto
web mvc all --package ~.web
entity jpa --class ~.domain.experiencia
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.experiencia
field number --fieldName idcurriculum --type java.lang.Integer --notNull
--class ~.domain.experiencia
field date --fieldName fecini --type java.util.Date --class
~.domain.experiencia
field date --fieldName fecfin --type java.util.Date --class
~.domain.experiencia
field number --fieldName salario --type java.lang.Integer --class
~.domain.experiencia
field string --fieldName tareas --class ~.domain.experiencia
web mvc all --package ~.web
// [failed] field number --fieldName id --type java.lang.Integer --
notNull --class ~.domain.oferta
entity jpa --class ~.domain.oferta
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.oferta
field string --fieldName titulo --class ~.domain.oferta
field string --fieldName infoPuesto --class ~.domain.oferta
field number --fieldName idFormacion --type java.lang.Integer --class
~.domain.oferta
field number --fieldName idSede --type java.lang.Integer --class
~.domain.oferta
field number --fieldName tipoContrato --type java.lang.Integer --class
~.domain.oferta
field number --fieldName salarioBruto --type java.lang.Integer --class
~.domain.oferta

```

```

field date --fieldName fechaInc --type java.util.Date --class
~.domain.oferta
field number --fieldName numVac --type java.lang.Integer --class
~.domain.oferta
field string --fieldName perfilDem --class ~.domain.oferta
field date --fieldName fecIniOferta --type java.util.Date --class
~.domain.oferta
field date --fieldName fecFinOferta --type java.util.Date --class
~.domain.oferta
field number --fieldName estado --type java.lang.Integer --class
~.domain.oferta
web mvc all --package ~.web
entity jpa --class ~.domain.tipoformacion
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.tipoformacion
field string --fieldName nombre --class ~.domain.tipoformacion
web mvc all --package ~.web
entity jpa --class ~.domain.formaciondemandante
field number --fieldName idformacion --type java.lang.Integer --notNull
--class ~.domain.formaciondemandante
field number --fieldName idcurriculum --type java.lang.Integer --notNull
--class ~.domain.formaciondemandante
field number --fieldName idcentro --type java.lang.Integer --notNull --
class ~.domain.formaciondemandante
web mvc all --package ~.web
entity jpa --class ~.domain.formacion
field number --fieldName id --type java.lang.Integer --notNull --class
~.domain.formacion
field number --fieldName idTipoFormacion --type java.lang.Integer --
notNull --class ~.domain.formacion
field number --fieldName titulacion --type java.lang.Integer --class
~.domain.formacion
field string --fieldName nivel --class ~.domain.formacion
web mvc all --package ~.web

//RELACIÓN DE CLAVES FORÁNEAS
field reference --fieldName idusuarioDem --type ~.domain.usuario --class
~.domain.demandante
field reference --fieldName idusuarioEmp --type ~.domain.usuario --class
~.domain.empresa
field reference --fieldName idDemandante --type ~.domain.demandante --
class ~.domain.inscripcion
field reference --fieldName idOferta --type ~.domain.oferta --class
~.domain.inscripcion
// [failed] field reference --fieldName idEmpresa --type

```



```

~.domain.empresa --class ~.domain.sede --cardinality ONE_TO_MANY
field reference --fieldName idEmpresa --type ~.domain.empresa --class
~.domain.sede --cardinality MANY_TO_ONE
// [failed] field reference --fieldName idSede --type ~.domain.sede --
class ~.domain.oferta --cardinality MANY_TO_ONE
field reference --fieldName idSedeOferta --type ~.domain.sede --class
~.domain.oferta --cardinality MANY_TO_ONE
field reference --fieldName idInscripcionOferta --type
~.domain.inscripcion --class ~.domain.oferta --cardinality MANY_TO_ONE
field reference --fieldName idInscripcionDemandante --type
~.domain.inscripcion --class ~.domain.demandante --cardinality
MANY_TO_ONE
field reference --fieldName idExperienciaDemandante --type
~.domain.demandante --class ~.domain.demandante --cardinality
MANY_TO_ONE
// [failed] field reference --fieldName idFormacionDemandanteFormacion -
-type ~.domain.formaciondemandante --class ~.domain.formacion --
cardinality MANY_TO_MANY
// [failed] field reference --fieldName idFormacionTipoFormacion --type
~.domain.formacion --class ~.domain.tipoFormacion --cardinality
MANY_TO_ONE
field reference --fieldName idFormacionTipoFormacion --type
~.domain.formacion --class ~.domain.tipoformacion --cardinality
MANY_TO_ONE
field reference --fieldName idOfertaFormacion --type ~.domain.oferta --
class ~.domain.formacion --cardinality MANY_TO_ONE
web mvc all --package ~.web
field reference --fieldName idPuestoOferta --type ~.domain.puesto --
class ~.domain.oferta --cardinality MANY_TO_ONE
field reference --fieldName idPuestoDemandante --type ~.domain.puesto --
class ~.domain.demandante --cardinality MANY_TO_ONE
field reference --fieldName idPuestoExperiencia --type ~.domain.puesto -
-class ~.domain.experiencia --cardinality MANY_TO_ONE
// [failed] field reference --fieldName idFormacionDemandanteFormacion -
-type ~.domain.formaciondemandante --class ~.domain.formacion --
cardinality MANY_TO_MANY
field reference --fieldName idFormacionDemandanteFormacion --type
~.domain.formaciondemandante --class ~.domain.formacion --cardinality
MANY_TO_ONE
field reference --fieldName idFormacionDemandanteDemandante --type
~.domain.formaciondemandante --class ~.domain.demandante --cardinality
MANY_TO_ONE
field reference --fieldName idFormacionDemandanteCentro --type
~.domain.formaciondemandante --class ~.domain.centro --cardinality
MANY_TO_ONE
// [failed] field reference --fieldName idPuestoOferta --type

```

```
~.domain.puesto --class ~.domain.oferta --cardinality MANY_TO_ONE
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-20
20:52:13
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-21
18:32:16
//IMPLANTACIÓN DE SEGURIDAD
security setup
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-21
20:00:44
web mvc all --package ~.web
//MULTILENGUAJE
web mvc language --code en
web mvc language --code es
//CREACIÓN DE BUSCADORES
finder add --finderName findOfertasByEstado --class ~.domain.oferta
web mvc finder all
finder add --finderName findOfertasBySede --class ~.domain.oferta
finder add --finderName findOfertasByIdSede --class ~.domain.oferta
finder add --finderName findOfertasByIdSede --class ~.domain.oferta
finder add --finderName findOfertasByTipoContrato --class
~.domain.oferta
finder add --finderName findOfertasByTitulo --class ~.domain.oferta
finder add --finderName findOfertasBySede --class ~.domain.oferta
finder add --finderName findOfertasByTituloLike --class ~.domain.oferta
finder add --finderName findEmpresasByNombreLike --class
~.domain.empresa
finder add --finderName findPerfilesByTrabajosDisp --class
~.domain.demandante
finder add --finderName findDemandantesByTrabajosDisp --class
~.domain.demandante
web mvc finder all
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-21
22:29:13
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-22
19:42:11
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-22
20:28:12
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-23
19:35:22
web mvc language --code de
//CREACIÓN DE TIPOS ENUMERADOS
enum type --class ~.inscripcion.estado
enum type --class ~.reference.estadoIns
enum constant --name Espera
enum constant --name Activada
```

```

enum constant --name Detenida
enum constant --name Cancelada
enum constant --name Tramite
enum constant --name Resuelta
focus --class ~.domain.inscripcion
field enum --fieldName estado --type ~reference.estadoIns --notNull
web mvc all --package ~.web
enum type --class ~.reference.tipoUsu
enum constant --name Empresa
enum constant --name Demandante
focus --class ~.domain.usuario
field enum --fieldName idTipoUsu --type ~reference.tipoUsu --notNull
enum type --class ~.reference.estadoOferta
enum constant --name Abierta
enum constant --name Cerrada
enum constant --name Cancelada
focus --class ~.domain.oferta
field enum --fieldName estadoOferta --type ~reference.estadoOferta --
notNull
web mvc finder all
web mvc all --package ~.web
// [failed] entity jpa --class ~.domain.demandante --sequenceName DE_SEQ
--testAutomatically
// [failed] entity jpa --class ~.domain.demandanteE --sequenceName
DEE_SEQ --testAutomatically
enum type --class ~.reference.tipoFormacion
enum constant --name ESO
enum constant --name Bachillerato
enum constant --name CFGM
enum constant --name CFGS
enum constant --name Universidad
enum constant --name Master
enum constant --name Doctorado
focus --class ~.domain.formacion
field enum --fieldName tipoForma --type ~reference.tipoForma --notNull
// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-23
21:19:30
// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-26
20:47:58
enum type --class ~.reference.tipoSexo
enum constant --name Masculino
enum constant --name Femenino
focus --class ~.domain.demandante
field enum --fieldName tipoSexo --type ~reference.tipoSx --notNull
web mvc all --package ~.web

```

```
web mvc all --package ~.web
enum type --class ~.reference.tipoContrato
enum constant --name Indefinido
enum constant --name Temporal
web mvc all --package ~.web
web mvc all --package ~.web
```

//PRUEBAS UNITARIAS

```
test mock --entity ~.domain.Centro
test mock --entity ~.domain.Demandante
test mock --entity ~.domain.Empresa
test mock --entity ~.domain.Experiencia
test mock --entity ~.domain.Formacion
test mock --entity ~.domain.Formaciondemandante
test mock --entity ~.domain.Inscripcion
test mock --entity ~.domain.Oferta
test mock --entity ~.domain.Puesto
test mock --entity ~.domain.Sede
test mock --entity ~.domain.Tipoformacion
test mock --entity ~.domain.Usuario
```

//PRUEBAS DE INTEGRACIÓN

```
test integration --entity ~.domain.Centro
test integration --entity ~.domain.Demandante
test integration --entity ~.domain.Empresa
test integration --entity ~.domain.Experiencia
test integration --entity ~.domain.Formacion
test integration --entity ~.domain.Formaciondemandante
test integration --entity ~.domain.Inscripcion
test integration --entity ~.domain.Oferta
test integration --entity ~.domain.Puesto
test integration --entity ~.domain.Sede
test integration --entity ~.domain.Tipoformacion
test integration --entity ~.domain.Usuario
```

//PRUEBAS DE WEB

```
selenium test --controller ~.web.CentroController
selenium test --controller ~.web.DemandanteController
selenium test --controller ~.web.EmpresaController
selenium test --controller ~.web.ExperienciaController
selenium test --controller ~.web.FormacionController
selenium test --controller ~.web.FormaciondemandanteController
selenium test --controller ~.web.InscripcionController
selenium test --controller ~.web.OfertaController
selenium test --controller ~.web.PuestoController
selenium test --controller ~.web.SedeController
selenium test --controller ~.web.TipoformacionController
selenium test --controller ~.web.UsuarioController
```

	// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-26 22:09:40
	// Spring Roo 1.3.2.RELEASE [rev 8387857] log opened at 2016-08-27 14:16:32
	web mvc all --package ~.web
	web mvc all --package ~.web
	// Spring Roo 1.3.2.RELEASE [rev 8387857] log closed at 2016-08-27 14:23:33

5.5. Interoperabilidad

Para cumplir el requisito no funcional de interoperabilidad, vamos a trabajar con AJAX para conseguir la carga dinámica de contenidos sin tener que realizar recargas de la página.

Para ello, será necesario descargar las librerías de AJAX, instalarlas, y realizar las dependencias. Una vez hecho esto, implementaremos mediante el lenguaje de programación JavaScript las correspondientes cargas de datos para aquellos apartados susceptibles de ser mejorados mediante tecnología AJAX.

El siguiente ejemplo, nos servirá como base para realizar las cargas dinámicas con AJAX.

Cuando se carga la página se ejecuta el método JavaScript que muestra el contenido de un archivo llamado `FaceJobs.txt` que se encuentra en el servidor. La clave del código anterior es que la petición HTTP y la descarga de los contenidos del archivo se realizan sin necesidad de recargar la página.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<title>Hola Mundo con AJAX</title>

<script type="text/javascript">
function descargaArchivo() {
    // Obtener la instancia del objeto XMLHttpRequest
    if(window.XMLHttpRequest) {
        peticion_http = new XMLHttpRequest();
    }
    else if(window.ActiveXObject) {
        peticion_http = new ActiveXObject("Microsoft.XMLHTTP");
    }

    // Preparar la funcion de respuesta
    peticion_http.onreadystatechange = muestraContenido;

    // Realizar peticion HTTP
```

```
peticion_http.open('GET', 'http://localhost/FaceJobs.txt', true);  
peticion_http.send(null);
```

```
function muestraContenido() {  
  if(peticion_http.readyState == 4) {  
    if(peticion_http.status == 200) {  
      alert(peticion_http.responseText);  
    }  
  }  
}  
}
```

```
window.onload = descargaArchivo;  
</script>
```

```
</head>  
<body></body>  
</html>
```