

Manual de instalación y cambios básicos en UDCDecide

UDCDecide es un Fork de la aplicación [cónsul](#), adaptada para su uso en la Universidad de A Coruña.

Este documento incluye información para la instalación desde cero de la plataforma en desarrollo y en producción, así como indicaciones para realizar cambios básicos en el portal.

Todos los comandos que se detallan se asumen lanzados desde la raíz del proyecto descargado.

1 Información

El fork de cónsul de la Universidad da Coruña está alojado en el CIXUG, en este repositorio: <https://git.cixug.es/osl/participacion>

En concreto las adaptaciones realizadas sobre cónsul se encuentran en la rama **udcdecide**, que se puede encontrar aquí: <https://git.cixug.es/osl/participacion/tree/udcdecide>

Para futuros desarrollos se puede:

- Seguir trabajando con esa rama. Asumiremos esta forma en el resto de instrucciones de desarrollo y despliegue.
- Hacer un "merge" a *master* y cambiar la configuración de despliegue para que sea la rama que se ponga en producción. Para ello editaremos la sección `branch` en la sección `production` del fichero `config/deploy-secrets.yml`

2 Instalación del sistema

Detallaremos los pasos necesarios para instalar el sistema en desarrollo.

Estas instrucciones asumen que se está utilizando un entorno Ubuntu Linux, en una versión

actualizada (2017 al menos).

Las instrucciones son válidas para configurar un entorno de producción y uno de desarrollo. En caso de existir diferencias se aclaran en cada punto.

Es importante no pasar al siguiente paso antes de haber completado los anteriores.

2.1 Software necesario

git

Si no estuviese instalado en el sistema, basta con ejecutar:

```
$ sudo apt-get install git
```

Para seguir este manual y para desarrollo y despliegue del sistema es necesario tener conocimientos de trabajo con *git*. Se manejarán comandos como *push*, *pull*, *merge* y trabajo con ramas (*branch*).

La [documentación oficial](#) es una buena fuente de aprendizaje en caso de no estar familiarizado con git.

ruby

- En producción instalarlo "standalone" ejecutando `$ sudo apt-get install ruby<version>` donde es la versión que se indica en el fichero `.ruby-version` de la raíz del proyecto. Si no se encontrase, instalar una versión de ruby2 ≥ 2.1 .
- En desarrollo es recomendable instalarlo usando el gestor de versiones de ruby [rbenv](#). Para ello, se instala primero rbenv y luego el ruby que se necesite (usa el fichero `.ruby-version` del que hablamos). Para ello:
 - i. Seguir los pasos indicados [aquí](#) hasta antes de "Install Ruby"
 - ii. Correr `$ rbenv install`, que instalará la versión configurada en `.ruby-version`

gem

Las gemas son los paquetes de software de ruby. Una vez instalado ruby lo que instalaremos a continuación serán gemas.

En principio sólo instalaremos `bundler` usando el comando `gem`, luego este gestor nos abstraerá el uso del comando `gem`.

bundler

Bundler es un gestor de dependencias ruby.

Nos permitirá instalarlas, configurar las que necesita nuestro software en un fichero `Gemfile` y ejecutarlas en un contexto cerrado. De no estar familiarizado con bundler, se recomienda la [documentación oficial](#).

Para instalarlo sólo tenemos que ejecutar:

```
$ gem install bundler
```

Los comandos que comienzan con `rake` y `rails` deberán ir precedidos de `bundle exec`.

postgresql

Si no estuviese instalado, podemos instalarlo ejecutando:

```
$ sudo apt-get install postgresql postgresql-contrib
```

Esto creará un usuario y contraseña por defecto, que se usará en la configuración de desarrollo.

En producción es muy aconsejable crear un usuario separado y configurarlo en la app (ver sección de instalación en servidor).

2.2 Repositorio

Para poder acceder al repositorio hay que contactar con el CIXUG, por ejemplo a través del adjunto de participación (participacion.vepeu@udc.gal).

Además, tendrás como mínimo que configurar git y una clave ssh para poder conectarte al repositorio. Estos conceptos básicos están cubiertos en el [README](#) de Gitlab así como en la documentación de Git ya comentada.

El repositorio se encuentra en <https://git.cixug.es/osl/participacion> y la rama que nos interesa

es *udcdecide*.

Una vez configurado git y configurada la clave en gitblab, para descargarlo y empezar a trabajar sobre la rama hacemos:

```
$ git clone http://git.cixug.es/osl/participacion.git
$ git checkout branch udcdecide
```

2.3 Instalación

Los pasos son iguales en desarrollo y en producción, indicándose en caso de que no sea así. Además, para opciones exclusivas de producción se añade una sección específica.

Para instalar el sistema, una vez instalado lo anterior y desde el directorio en el que nos hemos descargado el repositorio tenemos que:

2.3.1 Instalar las gemas necesarias

```
$ bundle install
```

2.3.2 Crear una base de datos para la aplicación

Para ello podemos usar el comando de postgres `createdb`.

```
$ createdb <NOMBRE_DE_LA_BD>
```

Si no lo estuviese, tenemos que `arrancar postgresql` previamente.

```
$ postgres -D /usr/local/pgsql/data
```

2.3.3 Crear los ficheros de configuración

Estos son ficheros especiales que contienen credenciales que no deben guardarse en el repositorio o que deben configurarse por entorno. Están en formato `yaml`.

Necesitaremos: `config/database.yml` , `config/secrets.yml` y `config/deploy-secrets.yml` . Podemos comenzar con las plantilla que se proporcionan en `config` y que tienen el mismo nombre acabado en `.example` o los ejemplos siguientes, donde se extrae sólo lo necesario.

Hay que tener en cuenta que deberemos cambiar las variables que se indican entre `<VARIABLE>` .

config/database.yml

Usado por la aplicación para conectarse a la base de datos adecuada.

Se necesita una versión adaptada con en `<NOMBRE_DE_USUARIO>` , `<CONTRASEÑA>` y `<NOMBRE_DE_LA_BD>` tanto en desarrollo como en producción.

```
default: &default
  adapter: postgresql
  encoding: unicode
  host: localhost
  pool: 5
  username: <NOMBRE_DE_USUARIO>
  password: <CONTRASEÑA>

development: &development
  <<: *default
  database: <NOMBRE_DE_LA_BD>

production:
  <<: *development
```

config/deploy-secrets.yml

Usado por el comando de despliegue `cap deploy` (configurado en `config/deploy.rb`). Aquí se incluyen los datos actuales del servidor en el que está desplegada la app.

Este fichero sólo es requerido en el entorno de desarrollo

```
production:
  deploy_to: "/home/participa"
  branch_to_deploy: udcdecide
  ssh_port: 22
  server: 10.8.16.195
  user: participa
  full_app_name: udcdecide.udc.gal
```

config/secrets.yml

Aquí se configura el token que se usa para verificar las cookies, además de algunas opciones de configuración, como la url del CAS.

En este caso es el resultado de ejecutar `$ bundle exec rake secret` desde la raíz del proyecto, generaremos uno distinto para cada TOKEN.

```
default: &default
  secret_key_base: <TOKEN_SECRETO>

development:
  <<: *default

test:
  <<: *default

production: &production
  secret_key_base: <TOKEN_SECRETO>
  cas_server: 'cas.udc.es'
```

2.3.4 Crear la base de datos

Para ello creamos la base de datos y corremos las migraciones:

```
bin/rake db:create
bin/rake db:migrate
```

En desarrollo, metemos los datos iniciales de prueba de `dev_seed` :

```
bin/rake db:dev_seed
```

En producción hacemos lo mismo, pero usando las semillas en `seeds` :

```
bin/rake db:seed
```

Estos datos iniciales, semillas o *seeds*, se pueden configurar en `db/seeds.rb` y `db/dev-seeds.rb` .

2.4 Correr la app (desarrollo)

Usaremos el comando de rails `rails server`, llamándolo así arrancaremos el servidor en el puerto 3000:

```
$ bin/rails server
```

Si queremos ejecutarlo en un puerto concreto:

```
$ bin/rails server -p 4321
```

2.5 Instrucciones específicas para instalación en producción

2.5.1 Máquina de producción

La máquina de producción en la UDC es, actualmente la que se encuentra en:

- Nombre: udcdecide
- URL: <http://udcdecide.udc.gal>
- SO: Ubuntu 14.1 LST
- IP: 10.8.16.195
- Usuario con privilegios de sudo: `usuario`
- Usuario de deploy: `participa`

Solicitar passwords para los usuarios al [servizo de informática](#)

De no existir máquina de producción, consultar también con el [servizo de informática](#) para obtener una.

2.5.2 Configuración

La configuración es la misma que para la máquina de desarrollo, pero en las secciones de configuración que se indican, los datos se meten bajo 'desarrollo:'

Además, hay que instalar y configurar Passenger y Apache. Como se explica [aquí] (<https://www.phusionpassenger.com/library/install/apache/install/oss/trusty>).

2.6 Despliegue

Hay que estar en la red de la universidad, sino no se podrá realizar el despliegue (o conectarse a producción).

El script `config/deploy.rb` automatiza todo el proceso, así que sólo tendremos que ejecutar esta tarea de [capistrano](#):

```
$ bundle exec cap production deploy
```

Cuando acabe el comando, que puede tardar varios minutos, lo que tengamos en la rama `udcdecide` y empujado al repositorio de la UDC estará correctamente desplegado en <http://udcdecide.udc.gal>.

3 Cambios de datos en la plataforma

Explicaciones de algunos cambios que se pueden realizar en la plataforma.

Para realizar cualquiera de los cambios que se detallan es recomendable tener el sistema instalado en desarrollo.

Para ello hay que seguir los pasos de la [sección 2](#)

3.1 Pasos comunes para cambios en la plataforma

Los cambios que se indican en los puntos siguientes requieren haber conover el manejo del software que se lista en el punto 2.1.

En concreto, para poder realizar cualquier cambio se tiene que hacer:

1. Clonar el repositorio: `git clone git@git.cixug.es:osl/participacion.git`
2. Entrar en el directorio: `cd participacion`
3. Ir a la rama de udcparticipa: `git checkout udcdecide`
4. Seguir los pasos de Instalación del sistema del punto 2.3

5. Correr la aplicación en local, como se explica en el punto 2.4
6. Realizar los cambios que se consideren necesarios. Se pueden ir viendo en local (se explica en el cada punto dónde).
7. Añadir al commit, poner un mensaje descriptivo y empujar al repositorio remoto

```
git add .  
git commit -m "Añadidas condiciones a más información"  
git push
```

8. Desplegar en producción: `bundle exec cap production deploy`

Si ya tuviésemos el repositorio descargado y el sistema instalado en desarrollo bastaría con comprobar que no tenemos cambios locales, y, desde el directorio de participa hacer `git pull` o `git fetch` y `git merge` y continuar desde el punto 5.

3.2 Cambiar textos html

3.2.1 Más información / Política de privacidad / etc.

Son ficheros ERB (Embedded Ruby) que generan html. Se encuentran en `app/views/pages/`, en concreto los que nos interesan son:

- `app/views/pages/more_information.html.erb` -> "Más información"
- `app/views/pages/privacy.html.erb` -> "Política de privacidad"
- `app/views/pages/proposals_info.html.erb` -> "Información sobre as propostas"

Aunque estos ficheros pueden contener exclusivamente html también admiten cualquier tipo de notación en `ERB`.

Para enlazar a ellos desde cualquier otra página o plantilla ERB podemos usar la función auxiliar `page_path`. Por ejemplo un enlace a `more_information`:

```
<%= link_to "Texto del enlace", page_path('more_information') %></li>
```

Una vez realizados los cambios y guardado el fichero podemos ver el resultado (si tenemos el

servidor de desarrollo corriendo en localhost:3000) en:

- `http://localhost:3000/privacy`
- `http://localhost:3000/more_information ...`

Para que los cambios se hagan efectivos en producción habría además que hacer un commit con los cambios, hacer un push al repositorio y desplegar en producción. Por ejemplo, si cambiamos `more_information`:

```
$ git add app/views/pages/more_information.html.erb
$ git commit -m "Modificadas instrucciones de 'Más información'"
$ git push
$ bundle exec cap production deploy
```

3.2.2 Pie de página

El *footer* o pie de página se encuentra en `app/views/layouts/_footer.html.erb` y los pasos para modificarlo son los mismos que en el caso de las páginas de información.

3.3 Nombres de los centros

Los "centros" son simplemente etiquetas especiales de clasificación de elementos del portal. Cuando se actualizó el primer portal participativo de la UDC a la última versión de Cónsul todavía era un elemento novedoso, por lo que no se dispone de una herramienta de administración visual.

Para importar los centros "por defecto", extraídos del portal de la UDC hay una tarea rake que los crea o resetea a su estado original:

```
$ rake db:udc_geozones_seed
```

Si se quiere cambiar el nombre de un centro específico hay que actuar directamente sobre la base de datos. No se recomienda y **hay que tener mucho cuidado al eliminarlos** pues eliminaríamos los elementos relacionados.

La tabla en cuestión es 'geozones'. Podemos listar su contenido desde la consola de postgresql

```
consul_development=# select id, name from geozones;
```

id	name
2	Escola Técnica Superior de Arquitectura
3	Escola Técnica Superior de Enxeñeiros de Camiños, Canais e Portos
4	Escola Técnica Superior de Náutica e Máquinas
5	Escola Universitaria de Arquitectura Técnica
6	Facultade de Ciencias
7	Facultade de Ciencias da Comunicación
8	Facultade de Ciencias da Educación
9	Facultade de Ciencias da Saúde
10	Facultade de Ciencias do Deporte e a Educación Física
11	Facultade de Dereito
12	Facultade de Economía e Empresa
13	Facultade de Filoloxía
14	Facultade de Fisioterapia
15	Facultade de Informática
16	Facultade de Socioloxía
17	Escola Politécnica Superior
18	Escola Universitaria de Deseño Industrial
19	Escola Universitaria Politécnica
20	Facultade de Ciencias do Traballo
21	Facultade de Enfermaría e Podoloxía
22	Facultade de Humanidades e Documentación
23	Escola Universitaria de Enfermaría
24	Escola Universitaria de Relacións Laborais
25	Escola Universitaria de Turismo

(24 rows)

Y si queremos cambiar, por exemplo el nombre de la Facultade de Informática:

```
# update geozones set name = 'FIC' where id=15;
```

Se recomienda no eliminar ninguna, pero si se comprueba que no hay ningún elemento relacionado con la misma (en principio sólo debates, usuarios y propuestas podrían estarlo) o que no se quieren esos elementos podemos hacer:

```
# delete from geozones where id=15;
```

De ser una operación habitual se aconseja solicitar una herramienta para hacer estas operaciones más seguras y cómodas. Ya sea en forma de herramienta de línea de comandos o una opción en administración.

3.4 Borrar Usuarios / Bloquear usuarios

Los usuarios pueden borrar su cuenta desde su menú de usuario. Pulsando en la parte superior de la pantalla "A miña conta" y luego en "Dar de baixa".

Un **administrador** también podrá borrar usuarios que han sido bloqueados. Para ello podrá acceder a la sección de [usuarios bloqueados](#) donde se listan los usuarios que han sido bloqueados. Ahí verá un botón para eliminarlo en los botones de acción. Además podrá confirmar su bloqueo (simplemente lo quita de la cola de revisión) o cancelar el bloqueo.

Si se quiere eliminar un usuario del que no se ha solicitado bloqueo previamente hay que bloquearlo en esa sección de moderación > [bloquear usuarios](#).

Téngase en cuenta que:

- En cuanto se bloquea un usuario desde moderación el usuario no podrá volver a acceder a la web, y todas sus Propuestas/Debates/Comentarios serán ocultados y dejarán de ser visibles en la web.
- La confirmación del bloqueo por parte del administrador simplemente lo quita de la cola de revisión. Además ahí dispone de otras acciones, como cancelar el bloqueo o eliminar el usuario.
- Cuando se elimina un usuario se borra la información del usuario y sus identidades, por lo que el usuario podría volver a registrarse con el CAS.
- Si un usuario se bloquea no podría volver a registrarse con la misma identidad.

3.5 Otros textos

La mayoría de los textos de la aplicación están extraídos para facilitar su internacionalización.

Cuando en un fichero ERB se encuentre el heper `t`, por ejemplo `<%= t="`
`"admin.users.name"="" %=""` significa que en `config/locales` hay un fichero [yaml](#) que tiene esa cadena de texto.

Una buena opción para encontrar donde está definido es buscar por una de las cadenas, si queremos saber dónde se encuentra la cadena del texto anterior, por ejemplo, podemos buscar "admin:". La cadena en concreto se encontrará anidada bajo `admin` en alguno de los ficheros como sigue:

```
gl:
  admin:
    users:
```

```
name: "Nome"
```

Si no se sabe que cadena corresponde a un texto podemos optar por buscar la cadena. Una búsqueda de "Nome" en el directorio dará varios resultados, entre los que podemos encontrar el anterior.

4 Más información

Otro tipo de cambios de funcionamiento en la plataforma requieren conocimientos de rails y del sistema consul más en detalle.

Errores

Siempre con conocimiento de Rails y consul, ante errores del sistema se puede analizar el fichero de log, que está en `log/production.log` en producción y `log/development.log` en desarrollo.

Outros

De existir interés en algún tema concreto pueden contactar con participacion.vepeu@udc.gal.

_ Manual de administración

Desde esta sección (<http://udcdecide.udc.gal/admin>) puedes administrar el sistema, a través de las siguientes acciones:

> Temas de debate

Los temas (también llamados tags, o etiquetas) de debate son palabras que definen los usuarios al crear debates, para catalogarlos (ej: sanidad, movilidad, arganzuela, etc).

Aquí se pueden eliminar temas inapropiados, o marcarlos para ser propuestos al crear debates (cada usuario puede definir los que quiera, pero se le sugieren algunos que nos parecen útiles como catalogación por defecto; aquí se puede cambiar cuáles se sugieren)

> **Propuestas/Debates/Comentarios ocultos**

Cuando un moderador o un administrador oculta una Propuesta/Debate/Comentario aparecerá en esta lista. De esta forma los administradores pueden revisar que se ha ocultado el elemento adecuado.

Al pulsar "Confirmar" se acepta que se haya ocultado: se considera que se ha hecho correctamente.

Al pulsar "Volver a mostrar" se revierte la acción de ocultar y vuelve a ser una Propuesta/Debate/Comentario visible, en el caso de que se considere que ha sido una acción errónea el haberlo ocultado.

Para facilitar la gestión, arriba encontramos un filtro con las secciones: "pendientes" (los elementos sobre los que todavía no se ha pulsado "confirmar" o "volver a mostrar", que deberían ser revisados todavía), "confirmados" y "todos".

Es recomendable revisar regularmente la sección "pendientes".

> **Usuarios bloqueados**

Cuando un moderador o un administrador bloquea a un usuario aparecerá en esta lista. Al bloquear a un usuario, éste deja de poder utilizarlo para ninguna acción de la web. Los administradores pueden desbloquearlos pulsando el botón al lado del nombre del usuario en la lista.

> **Cargos Públicos**

En la web, los usuarios individuales pueden ser usuarios normales, o cargos públicos. Estos últimos se diferencian de los primeros únicamente en que al lado de sus nombres aparece una etiqueta que les identifica, y cambia ligeramente el estilo de sus comentarios. Esto permite que los usuarios les identifiquen más fácilmente.

Al lado de cada usuario vemos la identificación que aparece en su etiqueta, y su nivel (la manera que internamente usa la web para diferenciar entre un tipo de cargos y otros). Pulsando el botón "Editar" al lado del usuario, se puede modificar su información. Los cargos públicos que no aparecen en la lista pueden ser encontrados para actuar sobre ellos por medio del buscador en la parte superior.

> Moderadores

Mediante el buscador de la parte superior se pueden buscar usuarios, para activarlos o desactivarlos como moderadores de la web. Los moderadores al acceder a la web con su usuario ven en la parte superior una nueva sección llamada "Moderar"

>Actividad de moderadores

En esta sección se va guardando todas las acciones que realizan los moderadores o los administradores respecto a la moderación: ocultar/mostrar Propuestas/Debates/Comentarios y bloquear usuarios. En la columna "Acción" comprobamos si la acción corresponde con ocultar o con volver a mostrar (restaurar) elementos o con bloquear usuarios. En las demás columnas tenemos el tipo de elemento, el contenido del elemento y el moderador o administrador que ha realizado la acción. Esta sección permite que los administradores detecten comportamientos irregulares por parte de moderadores específicos y que, por lo tanto, puedan corregirlos.

> Configuración Global

Opciones generales de configuración del sistema.

La mayoría de las opciones de configuración no se deben tocar. Algunas no cambiarán nada que se pueda ver en udc-decide, pero otras pueden hacer que el portal deje de funcionar. Es importante no activar funcionalidades que no son las activadas por defecto, ya que no se han configurado ni probado en el fork.

Aquí podemos cambiar sin problema:

- Los nombres de distintos niveles de cargos públicos.
- "Porcentaje máximo de votos anónimos por debate".
- "Número de votos en que un Debate deixa de se poder editar".
- "Prefixo para os códigos de Propostas".
- "Número de votos necesarios para aprobar unha Proposta".
- "Código a incluir en cada páxina". Lo que pongamos aquí aparecerá en la parte superior de cada página del portal. Por ejemplo:

```
<div style="text-align: center; background-color: lightslategray;color: lightgray;f  
Mantemento da plataforma o Xoves ás 23.00 a web pode deixar de esatar operativa  
</div>
```

> Estadísticas

Aquí se muestran estadísticas generales del sistema sobre usuarios, propuestas, votos, etc.

_ Manual de moderación

> Propuestas / Debates / Comentarios

Cuando un usuario marca en una Propuesta/Debate/Comentario la opción de "denunciar como inapropiado", aparecerá en esta lista. Respecto a cada uno aparecerá el título, fecha, número de denuncias (cuántos usuarios diferentes han marcado la opción de denuncia) y el texto de la Propuesta/Debate/Comentario.

A la derecha de cada elemento aparece una caja que podemos marcar para seleccionar todos los que queramos de la lista. Una vez seleccionados uno o varios, encontramos al final de la página tres botones para realizar acciones sobre ellos:

- **Ocultar:** hará que esos elementos dejen de mostrarse en la web.
- **Bloquear autores:** hará que el autor de ese elemento deje de poder acceder a la web, y que además todos las Propuestas/Debates/Comentarios de ese usuario dejen de mostrarse en la web.
- **Marcar como revisados:** cuando consideramos que esos elementos no deben ser moderados, que su contenido es correcto, y que por lo tanto deben dejar de ser mostrados en esta lista de elementos inapropiados.

Para facilitar la gestión, arriba encontramos un filtro con las secciones:

- **Pendientes:** las Propuestas/Debates/Comentarios sobre los que todavía no se ha pulsado "ocultar", "bloquear" o "marcar como revisados", y que por lo tanto deberían ser revisados todavía
- **Todos:** mostrando todos las Propuestas/Debates/Comentarios de la web, y no sólo los marcados como inapropiados.
- **Marcados como revisados:** los que algún moderador ha marcado como revisados y por

lo tanto parecen correctos.

Es recomendable revisar regularmente la sección "pendientes".

> **Bloquear usuarios**

Un buscador nos permite encontrar cualquier usuario introduciendo su nombre de usuario o correo electrónico, y bloquearlo una vez encontrado.

Al bloquearlo, el usuario no podrá volver a acceder a la web, y todas sus Propuestas/Debates/Comentarios serán ocultados y dejarán de ser visibles en la web.