# A qui va dirigit

Aquest how-to va dirigit als perfils tècnics (desenvolupadors i arquitectes) que vulguin començar a treballar amb el nou entorn de treball de Canigó (màquina virtual generada mitjançant Vagrant) i/o desplegar una aplicació demo de Canigó 3.1.X a un contenidor Docker.

# Versió de Canigó/entorn de treball

Els passos descrits en aquest document apliquen a la versió del Framework Canigó 3.1.x (front-end amb Bootstrap) i a la versió 1.0.0 de l’entorn de treball.

# Introducció

Aquest how-to pretén guiar a l’usuari a través del procés de desplegament d’una aplicació Canigó 3.1.x a un contenidor Docker amb MySQL i Tomcat. Tot el procés es realitza des de la màquina virtual amb el nou entorn de treball de Canigó.

Com a pre-requisit per poder seguir aquest how-To, cal disposar de la màquina virtual amb l’entorn de treball ja muntat.

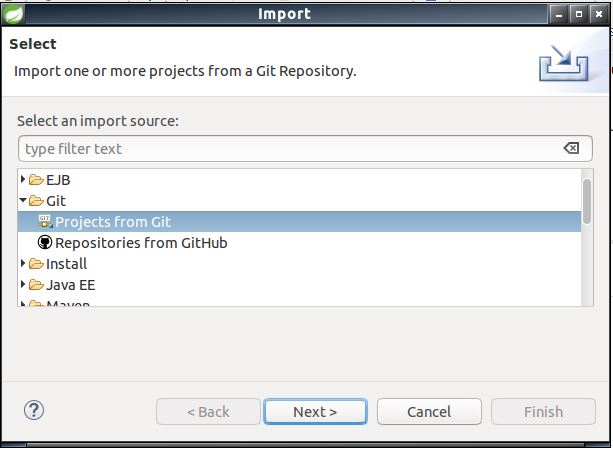
El codi font de l’aplicació demo es troba penjat al Git del cs-canigo i els recursos necessaris per desplegar-la (configuració del docker + scripts de BBDD) es troben inclosos al projecte

# Importació del projecte a l’entorn de treball

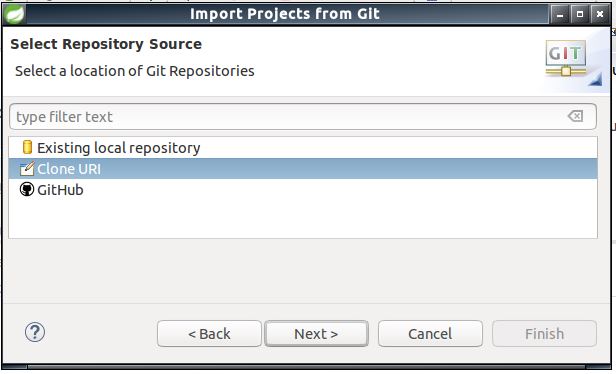
El primer pas és disposar del projecte a desplegar. El projecte-demo de Canigó anomenat ‘Equipaments’ amb el que treballarà aquest how-to es troba penjat al Git.

Es pot descarregar des de el propi Eclipse de l’entorn de treball:

* Seleccionar l’opció ‘Import’ del Menú superior de ‘File’.
* Escollir ‘Projects from Git’ com a tipus de recurs a importar.

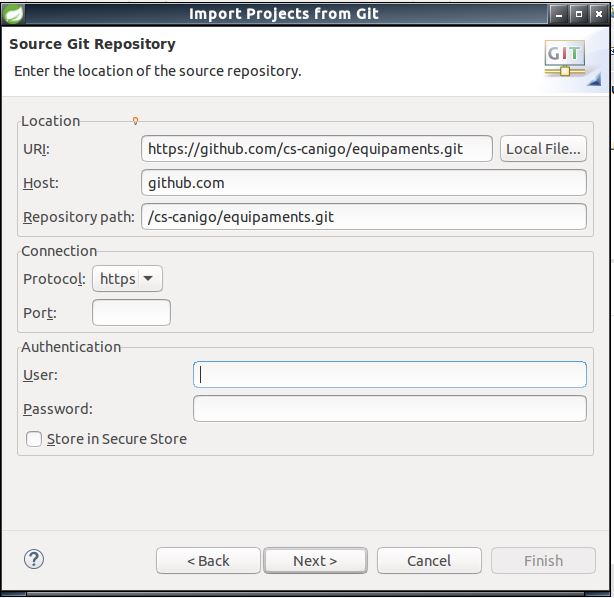


* Escollir ‘Clone URI’ a la següent finestra.

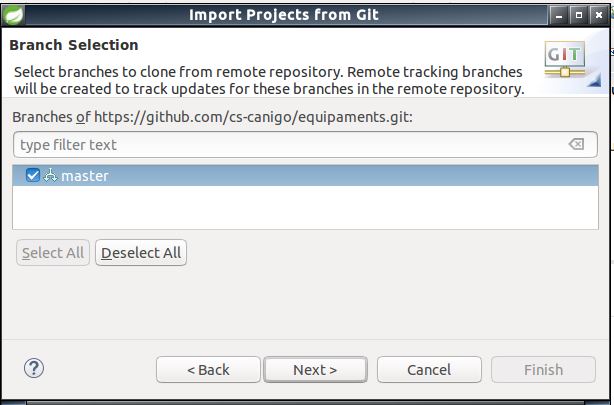


* Informar com a ‘UR’I: <https://github.com/cs-canigo/equipaments.git>

En fer-ho, s’ompliran automàticament els camps ‘Host’ i ‘Repositori path’.



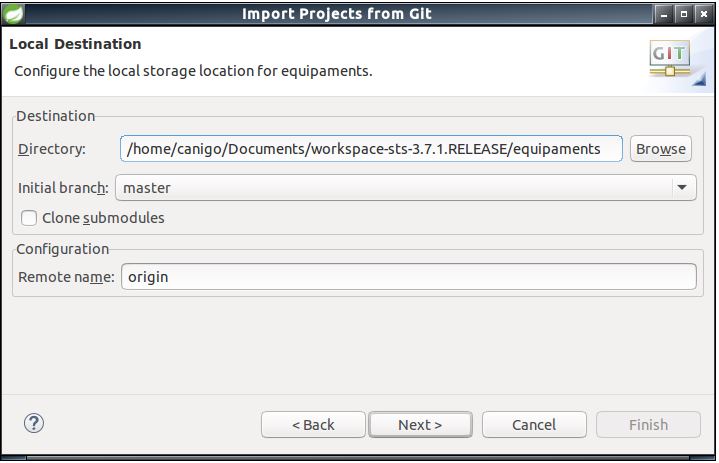
* Seleccionar ‘master’ com a branca a importar.



* Seleccionar el directori on descarregar el projecte.

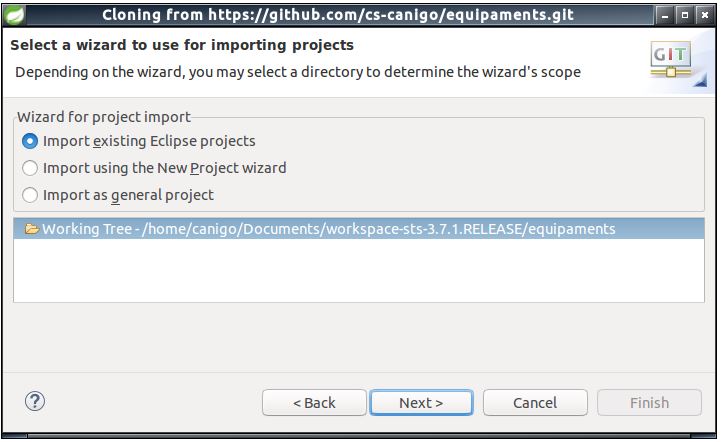
Serà important recordar aquest path donat que es referenciarà en el fitxer de configuració del Docker. A l’exemple que es treballa en aquest how-to es farà servir el path:

“/home/canigo/Documents/workspace-sts-3.7.1.RELEASE/equipaments”.

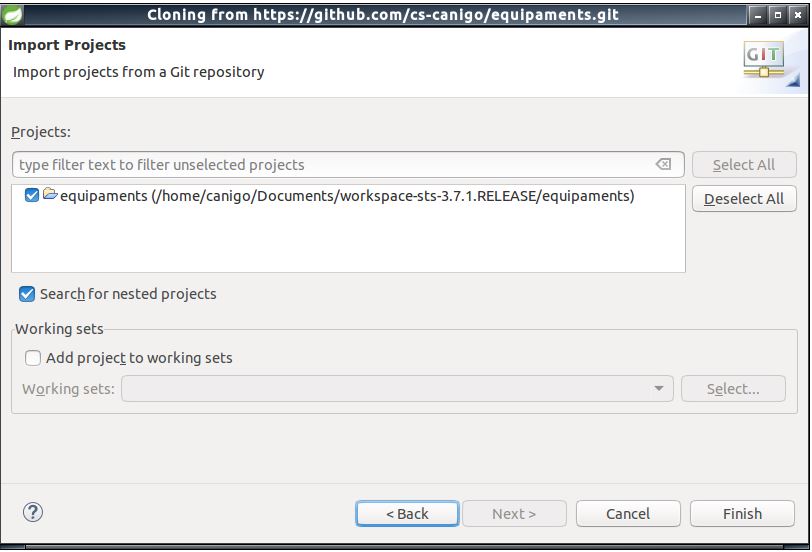


A la resta de camps, mantenir el valor per defecte.

* A l’assistent per importar projecte, seleccionar la opció de ‘Import existing Eclipse Project wizard’

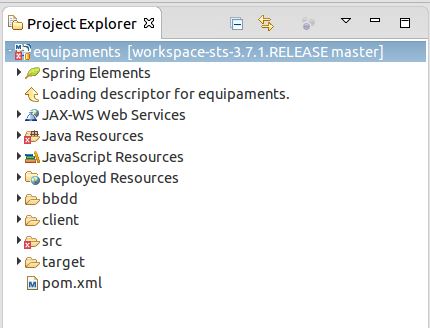


* Seleccionar ‘equipaments’ com a projecte a importar (l’únic):



Finalitzem l’assistent.

En acabar, tindrem descarregat el codi font de la demo de ‘Equipaments’:



# Configuració del Docker

Per a desplegar la demo d’Equipament es farà servir un Docker configurat amb el següent stack:

* MySQL 5.6: [mysql:5.6](https://hub.docker.com/_/mysql/)
* Tomcat 7 + Oracle JDK 7: [cscanigo/tomcat7-java7](https://hub.docker.com/r/cscanigo/tomcat7-java7/)
* Apache 2.4: [cscanigo/httpd:2.4](https://hub.docker.com/r/cscanigo/httpd/)

El fitxer amb la configuració del contenidor Docker es troba a:

.../src/main/docker/stacks/dev/apache-tomcat-mysql/docker-compose.yml

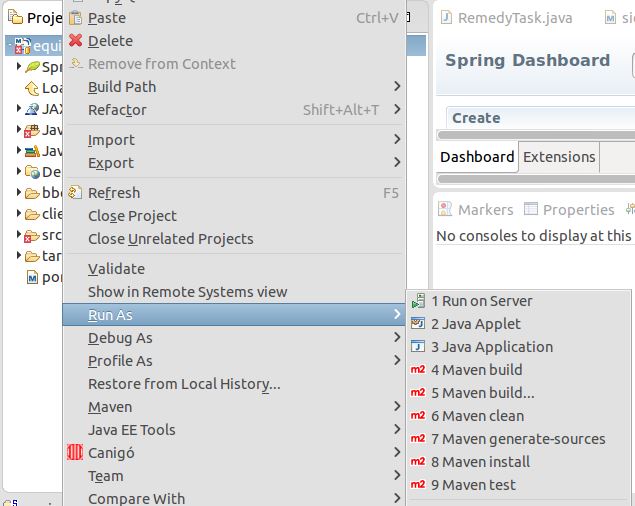
En aquest arxiu de configuració es referencien paths cap a recursos inclosos dins el projecte. En cas d’haver importat el projecte a un path diferent al que s’ha seguit en aquest how-to, caldrà actualitzar les referències:



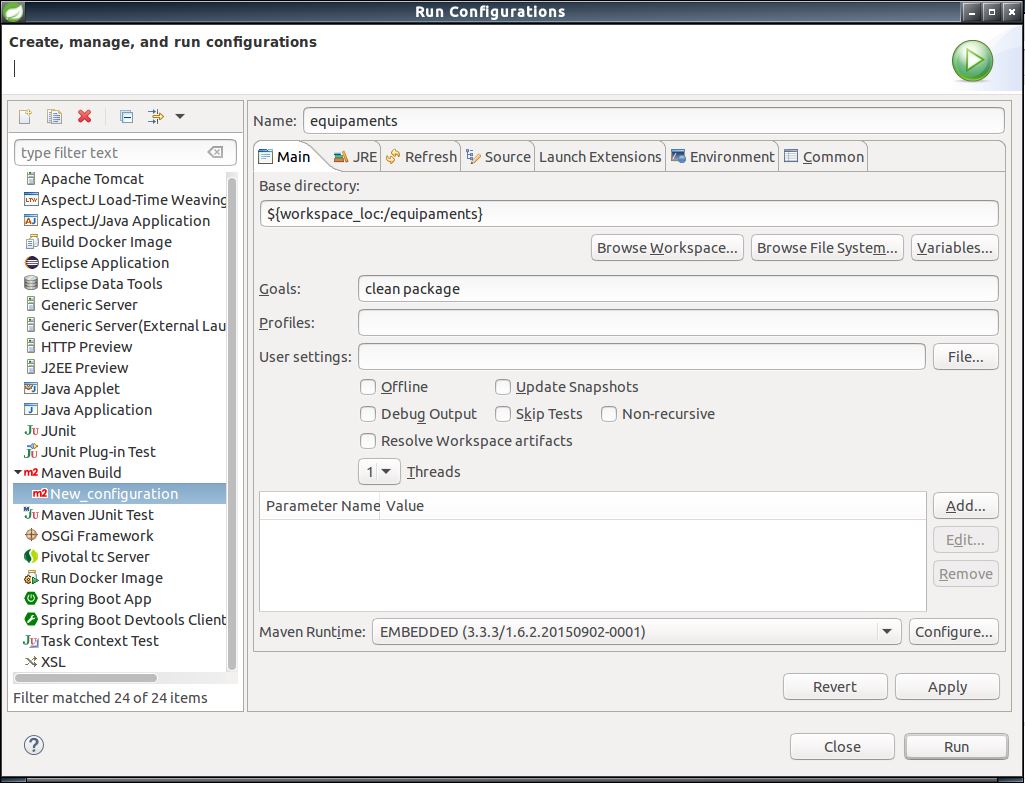
# Construcció de l’aplicació

Per tal de construir l’aplicació amb Maven mitjançant l’Ecplise cal:

* Crear una nova configuració amb Maven build



* Configurar la tasca:
  + Name: equipaments
  + Base directory: ${workspace\_loc:/equipaments}
  + Goals: clean Package



A la resta de camps, mantenir el valor per defecte. Una vegada configurat, cal aplicar canvis i executar.

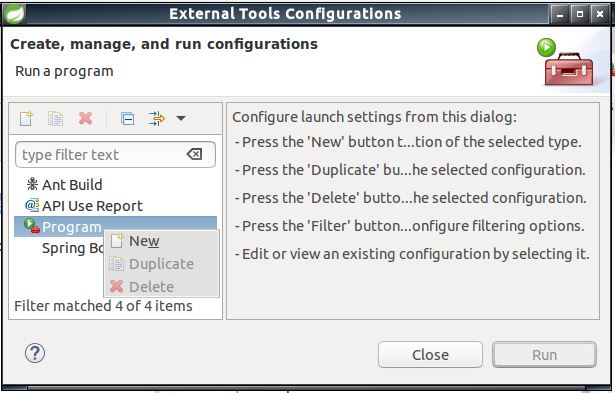
# Configuració de desplegament

Per tal de desplegar l’aplicació al contenidor Docker proporcionat cal:

* Seleccionar l’opció ‘External Tools’ del Menú superior de ‘Run’



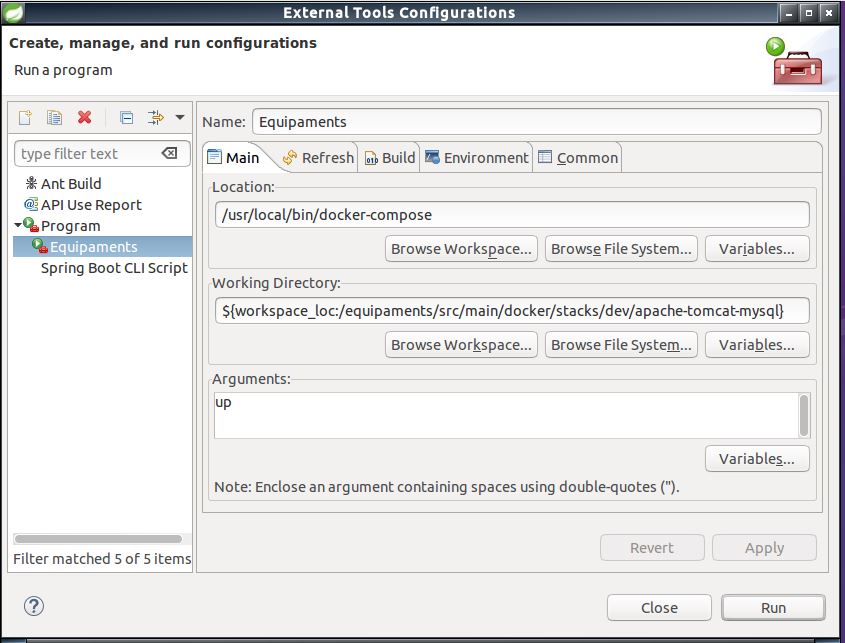
* Crear un nou ‘Program’



* Configurar la configuració amb:
* Name: Equipaments
* Location: /usr/local/bin/docker-compose
* Working Directory:

${workspace\_loc:/equipaments/src/main/docker/stacks/dev/apache-tomcat-mysql}

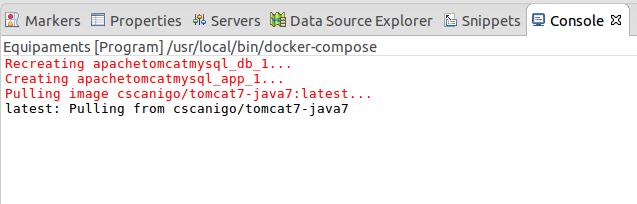
* Arguments: up



En cas que el projecte s’hagi desat sota un directori de treball diferent al que s’indica en aquest how-to, cal actualitzar la referència al camp ‘Working Directory’.

* Executar el desplegament (botó Run)

En fer-ho, a la consola hauria d’aparèixer el registre de creació del contenidor Docker



Si es desitja fer el desplegament via consola de comandes, es pot fer amb la següent comanda:

$ cd “/home/canigo/Documents/workspace-sts-3.7.1.RELEASE/equipaments /src/main/docker/stacks/dev/apache-tomcat-mysql/

$ docker-compose up

En cas que el projecte s’hagi desat sota un directori de treball diferent al que s’indica en aquest how-to, cal actualitzar la referència directori on es troba la configuració del contenidor Docker.

# Accés a l’aplicació desplegada

L’aplicació s’ha configurat per ser accessible mitjançant la següent URL:

<http://demos.canigo.ctti.gencat.cat/demo-equipaments>

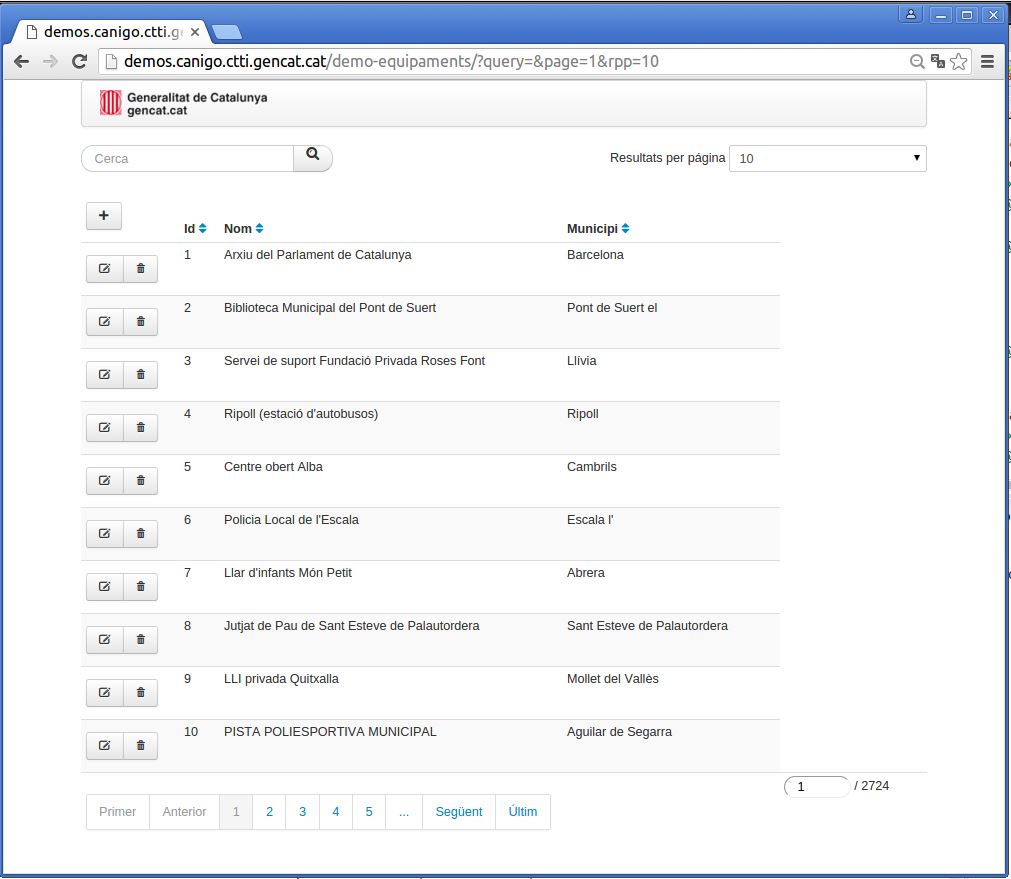
Per tal de que es reconegui el domini, caldrà incloure al fitxer /etc/hosts la següent entrada:

127.0.0.1 demos.canigo.ctti.gencat.cat

Donat que aquest fitxer es trova protegit, caldrà obrir-ho amb permisos d’administrador. Per exemple amb la següent comanda (emprant el editor de text ‘leafpad’):

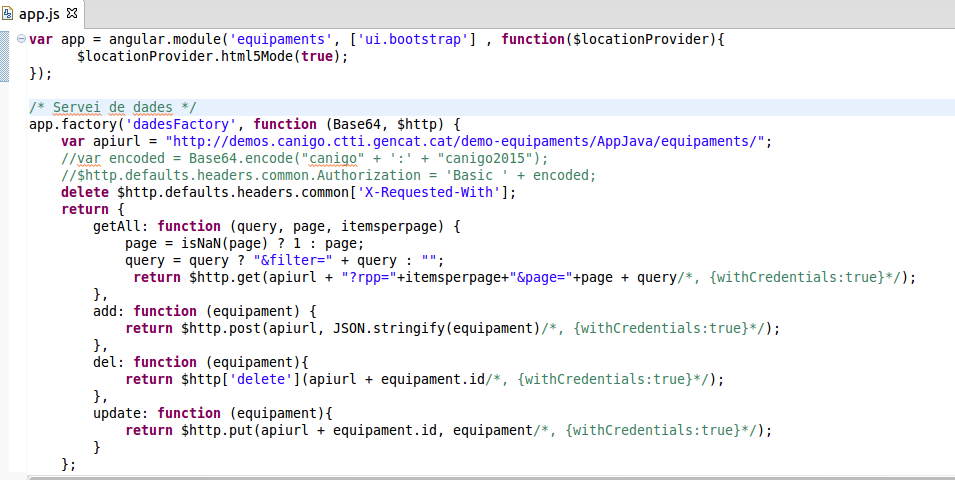
$ sudo leafpad /etc/hosts

L’aplicació presenta el següent aspecte:



En cas de voler fer servir un altre nom de domini (o accedir directament per ‘localhost’ per prescindir del canvi al fitxer hosts), cal modificar aquest nom de domini al següent fitxer dins el projecte:

/equipaments/client/js/app.js



# Eliminar contenidors en execució

La forma més fàcil d’eliminar els contenidors Docker en execució, i per tant aturar l’aplicació, és executar la següent comanda:

$ docker rm -f $(docker ps -qa)