

Instalación de ambiente Wissen

Instalar NodeJS

Desde la pagina de node: <https://nodejs.org/en/download/>

Instalar NVM (es un manejador de versiones para nodeJS)

Descargar: <https://github.com/coreybutler/nvm-windows/releases/latest>

Seguir el instructivo de:

<https://danielarancibia.wordpress.com/2017/03/28/install-or-upgrade-nodejs-with-nvm-for-windows/>

Esto será necesario ya que con la última versión de node, las aplicaciones no funcionan

Para hacer un downgrade ejecutaremos:

```
nvm install 6.4.0
```

Luego:

```
nvm use 6.4.0
```

Instalar Python para node

Fuente: <https://stackoverflow.com/a/39648550/9300379>

Abriendo la consola con permisos de administrador, CMD no powershell...
ejecutamos:

```
npm install -g --production windows-build-tools
```

```
npm install -g node-gyp
```

Instalar Grunt y Bower

```
npm install -g bower grunt grunt-cli
```

Con esto tenemos lo necesario para levantar el FrontEnd

Podemos levantarlo apuntando a otro ambiente o configurar el resto de la aplicación y usarlo local

Descargar código

Repositorio GIT

<http://10.200.0.125/Wissen/wissen>

Estructura del proyecto:

Carpeta wissen-backend: backend

Carpeta wissen-frontend: frontend

Para ejecutar el back end

Con Eclipse:

Importar proyecto maven.

Crear nuevo Run Configuration

Main class:

com.sc.wissen.Application

Program arguments:

--spring.profiles.active=desa

Para que use los datos de desarrollo tomando la config de src/main/resources/config/application-desa.yaml

Para Front-end

Configuración inicial

Los pasos anteriores son de configuración de las herramientas de cada uno, los pasos a continuación se deberán ejecutar una vez por aplicación, con la consola ubicados en el path frontend del proyecto.

Para que se instalen las dependencias del proyecto ejecutar:

npm install

bower install

Ejecución

grunt serve:desa consumiría los servicios de desarrollo,

grunt serve:local los locales

Instalar herramientas para desplegar el BackEnd

Instalar JDK 8

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Instalar Eclipse

Eclipse IDE for Java Developers (32 o 64 bits según tu OS)

<https://www.eclipse.org/downloads/>

Addons que usamos:

Spring Tools (opcional): ayuda para Spring

JBPMN2: (opcional): editor de los flujos de BPM, no se modifican hace tiempo.

Instalar Maven

Estamos usando la última versión

<https://maven.apache.org/download.cgi>

Agregar lo siguiente en **\${user.home}/.m2/settings.xml** confirmar que eclipse está apuntando al mismo archivo (windows-> preferences-> maven-> user settings)

```
<servers>
  <server>
    <id>nexus-snapshots</id>
    <username>admin</username>
    <password>admin123</password>
  </server>
  <server>
    <id>nexus-releases</id>
    <username>admin</username>
    <password>admin123</password>
  </server>
</servers>

<mirrors>
  <mirror>
    <id>nexus</id>
    <mirrorOf>*</mirrorOf>
    <url>http://10.200.1.53:8081/repository/maven-public/</url>
  </mirror>
</mirrors>
```