

GRAILS - 1

APLICACIÓN DE PRUEBA

1) Crear un nuevo proyecto → New → Grails Project

Creación del controlador: New → Controller (estando sobre el proyecto)

En el caso de llamar al controlador listProjects, se habrá creado un archivo ^(clase) llamado listProjectController dentro de:

Nombre Proyecto

↳ controller

↳ nombre proyecto (package)

↳ listProjectController.groovy

Cuyo método "index" es el que se ejecuta por defecto.

2) Una vez se de a "Run as" se nos dará una dirección similar a:

http://localhost:8080/Nombre Proyecto

Para que todo funcione OK hay que escribir realmente:

http://localhost:8080/Nombre Proyecto/listProject * pierde la mayúscula

↳ se mantiene la mayúscula

↳ nombre del controlador

Si se quiere utilizar otro método que no sea el index (por defecto), debe definirse el método:

```
def metodoX = { }
```

y llamarlo en el navegador: http://localhost:8080/Nombre Proyecto/listProject

↳ metodoX

llamar a otro método para que se ejecute:

redirect (action: metodo2)

El render admite etiquetas HTML. Ejemplo:

render "<h1> Proyecto de pruebas </h1>"

Creación de la vista: New → Groovy Server Page (GSP) (Teniendo seleccionada el proyecto)
la vista se podrá localizar en: Nombre Proyecto/grails-app/views/listProjects
Nombre Proyecto views listProjects Nombre Pagina - GSP = Nombre Metodo ^{minúscula}

Pasando valores a la vista:

- Se crean dos variables:

def projectName = "Proyecto"

def dueDate = "3/3/2014"

La página se debe llamar igual que el método al que va ligado.

- Se envían las variables a la página

[project: projectName, date: dueDate] → equivalente a hacer un return

- En la página las variables se recuperan como:

(se puede poner el return delante)

`#{project}` y `#{date}`

Si en el método se hace un render => NO se cargará la página GSP.

"

~~práct~~

"

Creación del Modelo de datos: New → Domain Class

(se debe llamar como el controlador → listProjects)

- Los atributos que se pongan en el modelo son los que aparecerán en la db =>
Hay que especificar el tipo de variables: strings, date, etc...

- DB: conf → Data Source. groovy

en environments {
development {

en dbcreate poner "update" ←

en url eliminar "mem"

- Entrar a manipular la base de datos desde el navegador: (DB console)

http://localhost:8080/ProjectTracker/dbconsole

Aquí el apartado URL hay que cambiarlo por:

jdbc:h2:devDb (Hasta lo que hay hasta el primer ; de url en DataSource. groovy)

HAY QUE RECONSTRUIR EL PROYECTO PARA QUE EN EL DBCONSOLE APAREZCA

LA BASE DE DATOS LIST-PROJECTS

- Para editar la tabla: select * from list-projects y dar luego al botón de edit