

1. Fundamentos

Groovy y Grails

Groovy es un lenguaje de programación dinámico que se ejecuta en la JVM (Java Virtual Machine). Es un lenguaje de programación orientado a objetos que se utiliza principalmente para escribir scripts y aplicaciones de automatización.

Por otro lado, GSP (Groovy Server Pages) es un lenguaje de plantillas que se utiliza para generar contenido HTML dinámico en aplicaciones web de Grails. Los archivos GSP se utilizan para definir la estructura y el diseño de las páginas web, mientras que los archivos Groovy se utilizan para definir la lógica de la aplicación.

Los archivos .groovy contienen código Groovy que se ejecuta en el servidor, mientras que los archivos .gsp contienen código GSP que se utiliza para generar contenido HTML dinámico en el navegador del usuario.

```
String mensaje = "Hola, mundo!"
println mensaje
```

```
<h1>${mensaje}</h1>
<h1>Hola, mundo!</h1>
```

Grails es un framework de aplicaciones web que se ejecuta en la plataforma Java. Está diseñado para ser altamente productivo y utiliza el lenguaje de programación Groovy. Grails utiliza el patrón de arquitectura Modelo-Vista-Controlador (MVC).

MVC

MVC es un patrón de arquitectura de software que se utiliza comúnmente en el desarrollo de aplicaciones web. MVC significa Modelo-Vista-Controlador y se divide en tres componentes principales:

Modelo: representa los datos y la lógica de negocio de la aplicación. El modelo es responsable de interactuar con la base de datos y proporcionar los datos necesarios a la vista. Aca vamos a definir clases de dominio, que atributos van a tener, de que tipo y otras configuraciones.

Vista: representa la interfaz de usuario de la aplicación. La vista es responsable de mostrar los datos al usuario y de proporcionar una forma de interactuar con la aplicación. Estos son nuestras archivos .gsp

Controlador: actúa como intermediario entre el modelo y la vista. El controlador es responsable de recibir las solicitudes del usuario, procesarlas y enviar los datos necesarios al modelo y a la vista. Va a ser el encargado de recibir y enviar la informacion a la vista

Servicios: Es donde vamos a realizar la mayor parte de la logica y es la capa que se conecta con la base de datos para manipular la data