Aplicaciones con Enfoque Orientado a Servicios

Tema Nº13:Reglas de negocio

Indicador de logro Nº13:Implementa reglas de negocio mediante JBoss Rules (Drools)

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº13:**

Reglas de negocio

**Subtema 13.1:**

¿Qué es una regla de negocio?

Una regla de negocio es una condición que se debe satisfacer cuando se realiza una actividad de negocio. Una regla puede imponer una política de negocio, tomar una decisión o inferir nuevos datos de datos existentes.

**Ejemplos:**

**Regla de negocio para cotizar precios determinados para los clientes en función del estado del cliente**

Un proveedor establecer el estado preferido de cada cliente y crea una regla de negocio para determinar cuánto cuesta a cada cliente el mismo objeto. A continuación, se indican términos básicos que definen los bloques de creación de cada regla de negocio:

* **Condiciones**

Una condición describe una situación o estado que se debe colocar en orden para que ocurra una acción determinada. En este ejemplo, la condición es el estado del cliente.

* **Acciones**

Una acción es el suceso resultante de la evaluación de la condición. Cada acción se enlaza únicamente con la condición que le precede. En este ejemplo, la acción es la cotización del precio; es diferente para cada una de las condiciones que se encuentran.

|  |  |
| --- | --- |
| Estado del cliente | Precio de cotización |
| Bronce | S/ 5 |
| Plata | S/ 4 |
| Oro | S/ 3 |

La interacción entre las condiciones y las acciones determina la forma de la regla de negocio. Si la condición se cumple, se realiza la acción.

**Subtema 13.2:**

Drools Workbench

En la mayoría de las empresas la lógica de negocio se encuentra dispersa por diferentes sitios:

* En el código de las aplicaciones
* Hojas de cálculo
* En la mente de los expertos en la materia

Este hecho hace que, a la mayoría de las personas de la organización les sea complejo consultar y comprender las reglas que constituyen la base del negocio.

Los sistemas de gestión de reglas de negocio (BRMS) como Drools surgen ante la necesidad de centralizar y gestionar la lógica de negocio. Para ello, se codifica dicha lógica en forma de reglas de negocio, que sirven para tomar decisiones dentro de un contexto.

Las reglas se ejecutan dentro de un motor de reglas (BRM). Las reglas, por tanto, no sirven únicamente para representar la lógica de negocio, sino también para ejecutarla.

Además, el hecho de que la lógica se encuentre codificada en una regla, hace más fácil su comprensión que cuando se encuentra en el código de una aplicación, sobre todo para el personal no técnico.

**Ejemplos:**

1. Para crear un proyecto desde cero, seleccione la opción *File > New Project*, y seleccione la opción *Drools Project* de la sección *Drools*.

A screenshot of a cell phone

Description generated with very high confidence

1. Elija la opción de proyecto con ejemplo.

A screenshot of a cell phone

Description generated with very high confidence

1. Ingrese el nombre del proyecto, manteniendo la configuración de Maven y agregar un ejemplo de regla de negocio.

A screenshot of a cell phone

Description generated with very high confidence

1. Verifique la estructura del proyecto. El archivo DroolsTest.java contiene el código para ejecutar las reglas de negocio; Sample.drl contiene las reglas de negocio; y kmodule.xml contiene la configuración para lectura del paquete *rules*.

A close up of a map

Description generated with high confidence

1. El archivo Sample.drl contiene las reglas de negocio.

A screenshot of a social media post

Description generated with very high confidence

1. Ejecute DroolsTest como proyecto Java y verifique el resultado en la consola.

A screenshot of a cell phone

Description generated with very high confidence

**Subtema 13.3:**

Implementación de reglas de negocio con jBPM y JBoss Rules

A un servicio jBPM se le pueden agregar reglas de negocio.

**Ejemplos:**

El proyecto se continuará con el proyecto jBPM del tema 12.

1. Añada la clase Usuario.java dentro de la carpeta com.sample.

A screenshot of a cell phone

Description generated with very high confidence

1. En la carpeta *src / main / resources* agrega la carpeta com.rule. Añada el archivo **Rules.drl**.

A screenshot of text

Description generated with very high confidence

1. Añada la siguiente regla de negocio: solo el usuario *admin* puede ingresar al servicio expuesto por ESB Mule.

A screenshot of a cell phone

Description generated with very high confidence

1. Configure el archivo kmodule.xml para ubicar el paquete de reglas *com.rule*.

A screenshot of a cell phone

Description generated with very high confidence

1. Modifique la clase ControlService.java añadiendo el método *obtenerUrl* con la configuración del usuario **visitante**.

A screenshot of a social media post

Description generated with very high confidence

1. Modifique el método apiProductosFromMuleESB de la clase ControlService.java.

A screenshot of a social media post

Description generated with very high confidence

1. Ejecute el aplicativo y verifique el resultado de la consola.

A close up of a logo

Description generated with high confidence

1. Realice el cambio de usuario a **admin** en el método obtenerUrl.

A screenshot of a cell phone

Description generated with very high confidence

1. Ejecute el aplicativo y verifique el resultado de la consola.

A close up of a piece of paper

Description generated with high confidence

**Actividad:**

1. Cree un proyecto Drools similar al primer ejemplo y modifique las reglas para verificar otros resultados.
2. Realice una integración de Drools con jBPM y ESB Mule con al menos 3 ó 4 servicios gestionados por usuarios o establezca sus propias reglas.