

JBPM & Drools

With Telecom domain case study

Daywise Course Outline

Day 1

- * Unit 1 : jBPM Business Process
- * Unit 2 : jBPM Components Overview
- * Unit 3 : Writing and Using BPMN2
- * Unit 4 : jBPM Initial APIs

Day 2

- * Unit 5 : jBPM example using APIs
- * Unit 6 : jBPM Domain Specific Processes
- * Unit 7 : jBPM Human Interaction
- * Unit 8 : jBPM Persistence

Day 3

- * Unit 9 : Threading in jBPM
- * Unit 10 : jBPM Advanced Topics
- * Unit 11 : More jBPM Advanced Topics
- * Unit 12 : Debugging

Day 4

- * Unit 13 : Introduction to Drools
- * Unit 14 : Business Rules
- * Unit 15 : Drools Architecture
- * Unit 16 : Installing and configuring

Day 5

- * Unit 17 : Drools for Programmers
- * Unit 18 : A simple Drools applications
- * Unit 19 : Drools for Business Users
- * Unit 20 : Performance & Exception Handling

Detailed Outline

Unit 1 : jBPM Business Process

- * Introduction to Business Process
- * Structure of a Business Process
- * Define a Business Process using BPMN2
- * Execute a Business Process using jBPM APIs
- * Discuss selected Case Study elements and identify approach

Unit 2 : jBPM Components Overview

- * Components of the jBPM platform
- * How they fit together
- * Introduction to Business Process Management
- * Using jBPM components for BPM
- * Hands-on : Installing Eclipse & creating a new project

Unit 3 : Writing and Using BPMN2

- * Introduction to BPMN2

- * BPMN2 Elements
- * Using jBPM with BPMN2
- * Hands-on : Code demo to jBPM (how to handle and test different types of processes and sub processes)

Unit 4 : jBPM Initial APIs

- * API components for 5.4
- * API components for 6.0
- * Analyze test processes
- * Configure persistence
- * Work with human tasks

Unit 5 : jBPM example using APIs

- * Interact with external services
- * Processes and Sub-processes
- * Tests
- * Handling rule execution from processes
- * Creating processes that react from external events
- * Pass and check data between tasks of a process instance
- * Hands-on

Unit 6 : jBPM Domain Specific Processes

- * BPMN2 process files with external applications
- * jBPM runtime with external applications
- * Parameterizations for external interaction
- * Binding to the jBPM runtime.

- * Immediate and deferred external system interactions
- * Static vs Dynamic process
- * Hands-on

Unit 7 : jBPM Human Interaction

- * Human Tasks
- * Characteristics of human interaction
- * Standards to integrate human interaction into applications
- * Security injection
- * Task life cycle
- * Hands-on

Unit 8 : jBPM Persistence

- * jBPM standard persistence
- * Configure persistence
- * Using persistence
- * Managing persistence from the session perspective
- * Persistence & Transaction
- * Hands-on

Unit 9 : Threading in jBPM

- * Multithreading model
- * Scalability options
- * Execution Control
- * Hands-on

Unit 10 : jBPM Advanced Topics

- * Inheritance
- * Handling external notifications regarding process execution
- * Use the process designer
- * Using CDI in JBPM

Unit 11 : More jBPM Advanced Topics

- * Asynchronous processes
- * Looping and time event
- * Tracing the process flow and how can we re-trigger the process back
- * Understanding the concept of Work Item
- * Hands-on

Unit 12 : Debugging

- * Debug process executions
- * Identify errors and unexpected behaviour
- * Managing jBPM process repository
- * Logging
- * Exception Handling
- * Hands-on

Unit 13 : Introduction to Drools

- * Introducing Drools/JBoss Rules
- * Declarative Programming
- * Rule Engines, Expert Systems and Expert Shells
- * Forward and Backward Chaining
- * Why and when to use Rule Based Systems

Unit 14 : Business Rules

- * Implementing Business Rules
- * Designing a Rules Based system
- * The Rete Algorithm and ReteOO

Unit 15 : Drools Architecture

- * Architecture of a Rule Based System
- * Some real world applications of a Rule Engine
- * Drools Architecture
 - * RuleBase
 - * WorkingMemory
 - * Fact Manipulation
 - * Rule Execution

Unit 16 : Installing and configuring

- * Installing and configuring Drools in a JSE environment

Unit 17 : Drools for Programmers

- * The DRL File
- * The Drools Rule Language
- * Assertion, Retraction and Modification
- * Property Change Listeners
- * Globals
- * Initial and Shadow Facts
- * Agenda
- * Basic Conflict Resolution

Unit 18 : A simple Drools applications

- * Recipe Finder Application: A simple Drools applications
- * Using the Drools Eclipse IDE Plugin - Rule Workbench
- * Taking the Recipe Finder to the Web
- * Drools in Tomcat
- * Storing and Managing your Rules

Unit 19 : Drools for Business Users

- * Authoring Rules
- * DSLs - Domain Specific Languages
- * Decision Tables
- * Stateless and Stateful Sessions
- * Drools Event Model
- * Understanding the Agenda Rule Set for:
 - * executing control
 - * matching rules and control
 - * resolution of control conflict

- * working with the Decision Table and conditions for using it
- * Rule Set and Rule Table attributes

Unit 20 : Performance

- * Querying the Working Memory
- * Business Process Management with RuleFlows
- * Exception Handling
- * Performance Considerations

Telecom Domain Use Case:

- The case study deals with implementation of a rule based process for 4G activation request:
- Activation
 1. When a customer requests for 4G activation (from Frontend ordering channel),
 - request comes through Message Queue or Microservice
 - request is in XML format
 2. From the request, following steps need to be followed:
 - perform validation
 - determine the features that need to be provisioned
 - send the communication to specific systems based on the feature code
 3. If any exception occurs during above process:
 - retry and call the workflow again
 - Otherwise call the compensation logic to Revenue Assurance
- Change
 1. When a customer request for 4G activation (from Frontend ordering channel),
 - request comes through Message Queue or Microservice

- request is in XML format
2. From the request, following steps need to be followed:
 - perform rule validation
 - determine the features that need to be removed and added to the customer
 - send the communication in XML format to specific systems based on the feature code
 3. If any exception occurs during above process:
 - retry and call the workflow again
 - Otherwise we will call the compensation logic to Revenue Assurance
 4. Rules to be created for determining if it is activation or change request.