

Building Intelligent Applications with Kogito

PV207 BPM Labs

Mgr. Ivo Bek
Senior Product Manager

April 2022

Mgr. Marian Macik
Senior Quality Engineer



KOGITO

<https://kogito.kie.org/>

- Domain-Driven Development
- Generated Domain-specific APIs from your BPMN and DMN models
- Lightweight orchestration microservices
- Event-driven business logic
- Dev-mode hot reload
- Distributed sagas for microservices
- Serverless workflows
- Polyglot programming

More info at

- [Blogs](#)
- [Youtube](#)

PREREQUISITES

- Java 11 with JAVA_HOME configured
- [Visual Studio Code](#)
- [Kogito Bundle Extension](#) for VS Code
- Docker (only for Dev UI)

YOUR FIRST KOGITO APPLICATION

- Generate a preconfigured Quarkus application with Kogito extension at code.quarkus.io
 - Generate your application -> Download as a zip
 - Version without Dev UI available [here](#)
- Design BPMN/DMN in src/main/resources folder
 - VS Code
 - bpmn.new
- [myProcess](#)
- Start the application in development mode
 - `./mvnw clean compile quarkus:dev`
- Domain-specific API available at <http://localhost:8080/q/swagger-ui/>
 - OpenAPI specification
 - Generated only in development mode by default

YOUR FIRST KOGITO APPLICATION

Dev UI (only available with Docker)

- Start the process using Swagger UI
- Go to <http://localhost:8080/q/dev/> and select Process Instances under on the Kogito Runtime Tools card
- Look at the process in detail
- Same for the task, although it will be just listed, details will be unavailable due to a missing form

The screenshot displays the Kogito Dev UI interface. At the top, there is a header bar with the text "Dev UI" on the left and "my-first-kogito-application 1.0.0-SNAPSHOT (powered by Quarkus 2.8.0.Final)" on the right. Below the header, there are several tool cards:

- Configuration**: Contains "Config Editor" and "Dev Services".
- ArC**: Contains "Build time CDI dependency injection" and a list of components: "Beans 73", "Observers 4", "Interceptors 1", "Fired Events", "Invocation Trees", and "Removed Components 77".
- Kogito - Process (jBPM)**: Contains "Add Kogito Processes capabilities - Includes Process (jBPM) Engine" and "Data Index GraphQL UI".
- Kogito Runtime Tools**: Contains "Runtime development tools for Kogito projects" and a list of tools: "Process Instances 0", "Tasks 0", "Jobs 0", and "Forms 0". This card is highlighted with a red dashed border.
- SmallRye OpenAPI**: Contains "Document your REST APIs with OpenAPI - comes with Swagger UI" and "OpenAPI" and "Swagger UI".

YOUR FIRST KOGITO APPLICATION

- Change the greeting phrase
 - Edit an asset, e.g. the script task
 - Execute a command to invoke the Kogito endpoint again
 - Asset is recompiled and a new version is invoked
- Compile
 - `./mvnw clean install`
 - Append `-Dquarkus.swagger-ui.always-include=true` to include Swagger UI in production
- Run
 - `java -jar target/quarkus-app/quarkus-run.jar`
 - Application uses **whole quarkus-app** directory
- Run on a different port
 - `java -Dquarkus.http.port=8090 -jar target/quarkus-app/quarkus-run.jar`

YOUR FIRST KOGITO APPLICATION

"Raw commands"

- Start a process

```
curl -X POST -H 'Content-Type:application/json' -H 'Accept:application/json' -d '{"personName": "John"}'  
http://localhost:8080/myProcess
```

- Show processes

```
curl -H 'Accept:application/json' http://localhost:8080/myProcess
```

- Show Tasks

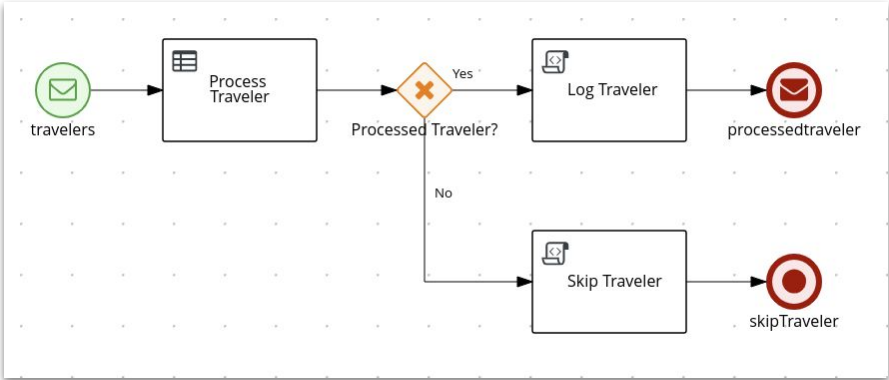
```
curl -H 'Accept:application/json' http://localhost:8080/myProcess/{puuid}/tasks?user=admin
```

- Complete a task

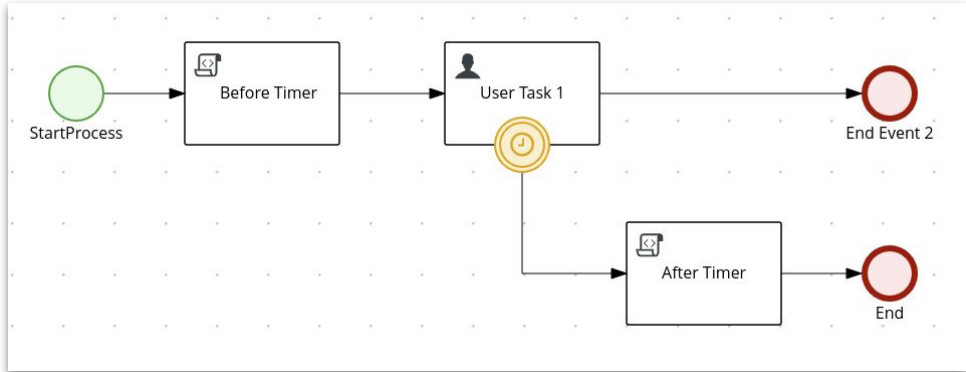
```
curl -X POST -H 'Content-Type:application/json' -H 'Accept:application/json' -d '{"personName": "John Doe"}'  
http://localhost:8080/myProcess/{puuid}/{taskName}/{tuuid}?user=admin
```

KOGITO EXAMPLES

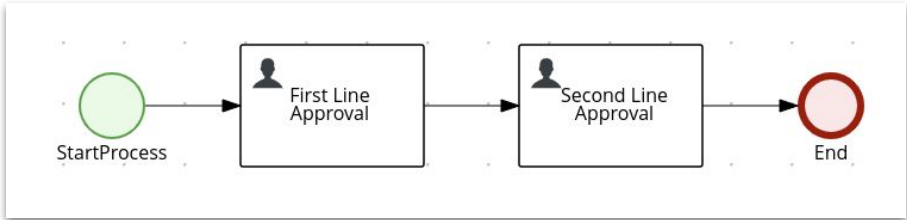
<https://github.com/kiegroup/kogito-examples>



KAFKA MESSAGE EVENTS
process-kafka-quickstart-quarkus



TIMERS
process-timer-quarkus



4-EYE PRINCIPLE
process-usertasks-quarkus

● ● ● and more

Thank you,
questions?