jBPM primer

Prasad Narwadkar

Table of Contents

[1. Overview 3](#_Toc96632244)

[1.1 Introduction to jBPM 3](#_Toc96632245)

[1.1 jBPM Architecture 5](#_Toc96632246)

[1.1.1 Core jBPM engine 5](#_Toc96632247)

[1.1.2 Web based tools 5](#_Toc96632248)

[1.1.3 Management Console 6](#_Toc96632249)

[1.1.4 Eclipse-based developer tools 6](#_Toc96632250)

[2. Installation and getting started 6](#_Toc96632251)

[2.1 Getting started 6](#_Toc96632252)

[2.1.1 Download jBPM 6](#_Toc96632253)

[2.1.2 Explore your running environment 7](#_Toc96632254)

[2.1.3 KIE Server Swagger documentation- jBPM RESTful API 8](#_Toc96632255)

[2.1.4 Try examples 9](#_Toc96632256)

[3. ASP.NET Core web app which works with a sample jBPM workflow using jBPM REST API 10](#_Toc96632257)

[3.1 Web app architecture 10](#_Toc96632258)

[3.2 The Business Process/Workflow 10](#_Toc96632259)

[3.3 UI Web app 11](#_Toc96632260)

[3.3.1 Web Application Roles Mapped to BPM users and roles 11](#_Toc96632261)

[3.3.2 Landing page of UI 12](#_Toc96632262)

[3.4 Go to My Evaluations (sample pre-built HR evaluation workflow/business process in jBPM) 13](#_Toc96632263)

[3.4.1 Start a new employee evaluation as a PM 13](#_Toc96632264)

[3.4.2 Login as jack@example.com to evaluate yourself 15](#_Toc96632265)

[3.4.3 Login as PM (a@b.com) again to do PM evaluation 16](#_Toc96632266)

[3.4.4 Login as HR (p@q.com) to evaluate an employee as HR 17](#_Toc96632267)

[4. References 19](#_Toc96632268)

# Overview

jBPM is an open-source BPM tool. It allows developers to use a RESTful API that enables users to do the same things (complete their tasks in the process) from within a Web App UI which they can do from the jBPM Business Central web application. It is extremely easy to install and run as you can see from the steps mentioned below.

## 1.1 Introduction to jBPM

jBPM is a very flexible Business Process Management (BPM) or Workflow management suite. It is light-weight, fully open-source and written in Java. Although, it is written in Java, it exposes a RESTful API and can hence be used in a headless manner from any client. It allows you to model, execute, and monitor business processes and cases throughout their life cycle.

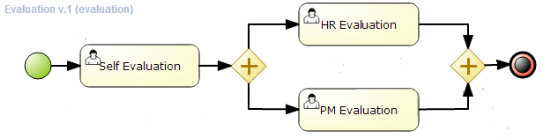


Figure Example process

A business process allows you to model your business goals by describing the steps that need to be executed to achieve those goals, and the order of those goals is depicted using a flow chart. This process greatly improves the visibility and agility of your business logic. jBPM focuses on executable business processes, which are business processes that contain enough detail so they can actually be executed on a BPM jBPM engine. Executable business processes bridge the gap between business users and developers as they are higher-level and use domain-specific concepts that are understood by business users but can also be executed directly.

Business processes need to be supported throughout their entire life cycle: authoring, deployment, process management and task lists, and dashboards and reporting.

The core of jBPM is a light-weight, extensible workflow engine written in pure Java that allows you to execute business processes using the latest [BPMN 2.0 specification](http://www.omg.org/spec/BPMN/2.0/). It can run in any Java environment, embedded in your application or as a service.

On top of the jBPM engine, a lot of features and tools are offered to support business processes throughout their entire life cycle:

* Pluggable human task service based on WS-HumanTask for including tasks that need to be performed by human actors.
* Pluggable persistence and transactions (based on JPA / JTA).
* Case management capabilities added to the jBPM engine to support more adaptive and flexible use cases
* Web-based process designer to support the graphical creation and simulation of your business processes (drag and drop).
* Web-based data modeler and form modeler to support the creation of data models and task forms
* Web-based, customizable dashboards and reporting
* All combined in one web-based Business Central application, supporting the complete BPM life cycle:
* Modelling and deployment - author your processes, rules, data models, forms and other assets
* Execution - execute processes, tasks, rules and events on the core runtime engine
* Runtime Management - work on assigned task, manage process instances, etc
* Reporting - keep track of the execution using Business Activity Monitoring capabilities
* Eclipse-based developer tools to support the modeling, testing and debugging of processes
* Remote API to jBPM engine as a service (REST, JMS, Remote Java API)
* Integration with Maven, Spring, OSGi, etc.

## jBPM Architecture

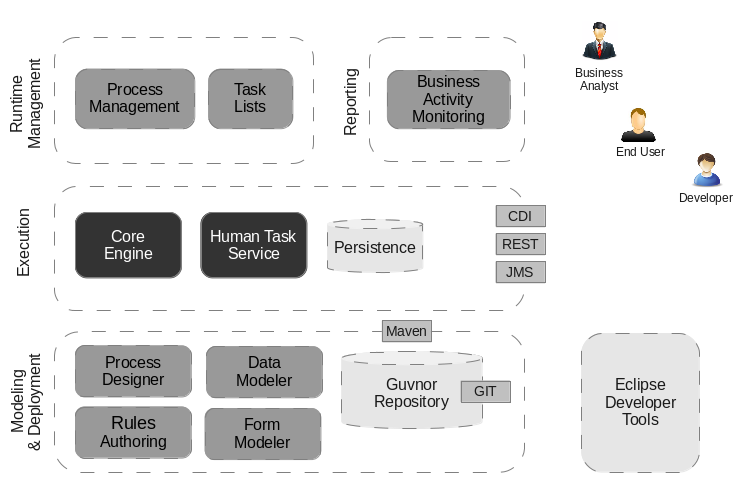


Figure jBPM architecture

This figure gives an overview of the different components of the jBPM project.

### Core jBPM engine

The core engine is the heart of the project and allows you to execute business processes in a flexible manner. It is a pure Java component that you can choose to embed as part of your application or deploy it as a service and connect to it through the web-based UI or remote APIs.

* An optional core service is the human task service that will take care of the human task life cycle if human actors participate in the process.
* Another optional core service is runtime persistence; this will persist the state of all your process instances and log audit information about everything that is happening at runtime.
* Applications can connect to the core engine through its Java API or as a set of CDI services, but also remotely through a REST and JMS API.

### Web based tools

These tools allow you to model, simulate and deploy your processes and other related artifacts (like data models, forms, rules, etc.):

* The process designer allows business users to design and simulate business processes in a web-based environment.
* The data modeler allows non-technical users to view, modify and create data models for use in your processes.
* A web-based form modeler also allows you to create, generate or edit forms related to your processes (to start the process or to complete one of the user tasks).
* Rule authoring allows you to specify different types of business rules (decision tables, guided rules, etc.) for combination with your processes.
* All assets are stored and managed by the Guvnor repository (exposed through Git) and can be managed (versioning), built and deployed.

### Management Console

The web-based management console allows business users to manage their runtime (manage business processes like start new processes, inspect running instances, etc.), to manage their task list and to perform Business Activity Monitoring (BAM) and see reports.

### Eclipse-based developer tools

The Eclipse-based developer tools are an extension to the Eclipse IDE, targeted towards developers, and allows you to create business processes using drag and drop, test and debug your processes, etc.

# Installation and getting started

Time to complete this: 5 minutes

## 2.1 Getting started

[See Using Single Zip Distribution to see details on how to install jBPM with the zip archive.](https://jbpm.org/learn/gettingStarted.html" \l "zip)

It is also possible to install jBPM on your machine/server [Using Docker.](https://jbpm.org/learn/gettingStarted.html#docker)

### Download jBPM

First things first, download jBPM server distribution (if not already done) to start quickly with complete environment.

[Download jBPM 7.65.0.Final server (single zip) distribution](https://download.jboss.org/jbpm/release/7.65.0.Final/jbpm-server-7.65.0.Final-dist.zip)

Just download, unzip and run

jbpm-server/bin/standalone.sh (Unix / Linux)

jbpm-server/bin/standalone.bat (Windows)

### Explore your running environment

Business Central provides feature rich authoring and management environment. It consists of:

* Design perspectives where users can create projects and pages
* Deploy perspectives where users can manage execution servers
* Manage perspectives where users can take control over process and case instances
* Track perspectives where users can access assigned tasks and reports

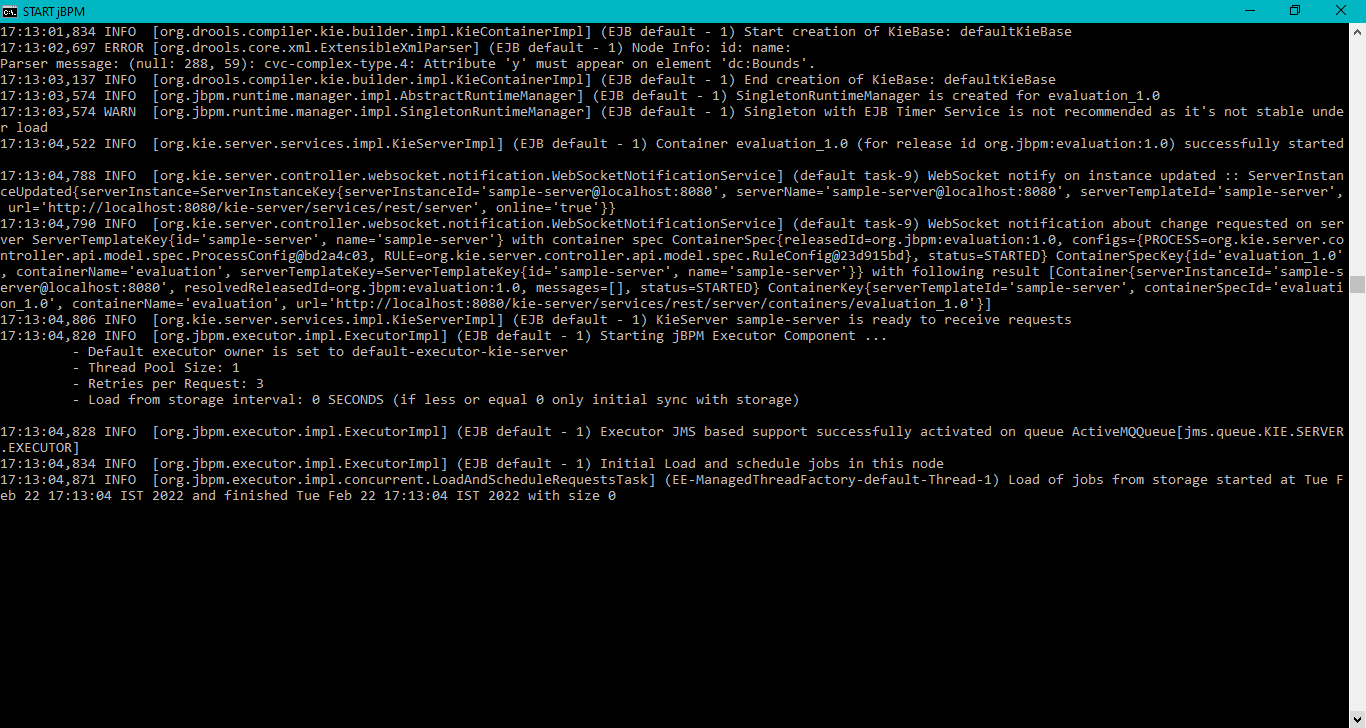


Figure jBPM engine running on port 8080 on my machine

Business Central can be accessed (once the server was started) at <http://localhost:8080/business-central>

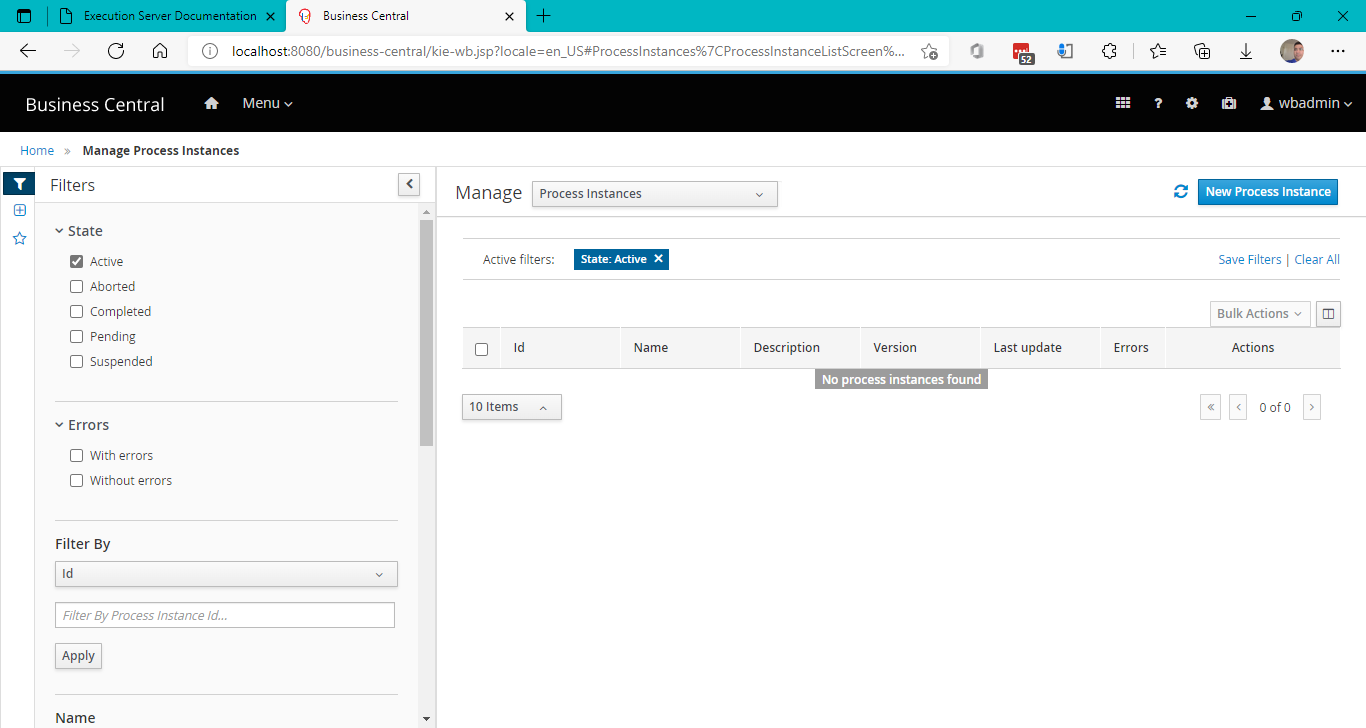


Figure Business Central where you can access your business processes and edit/deploy them

There are set of predefined users that can be used to directly logon to Business Central:

* wbadmin/wbadmin
* krisv/krisv
* john/john
* mary/mary
* katy/katy
* jack/jack
* kieserver/kieserver1!

Additional users can be created via Business Central Admin section.

### KIE Server Swagger documentation- jBPM RESTful API

KIE Server is the execution server that provides various capabilities

BRM (business rules)

BPM (business processes)

BRP (planning/solver)

Case Management

DMN (decisions)

Its complete REST api documentation can be accessed at the following uri.

<http://localhost:8080/kie-server/docs>

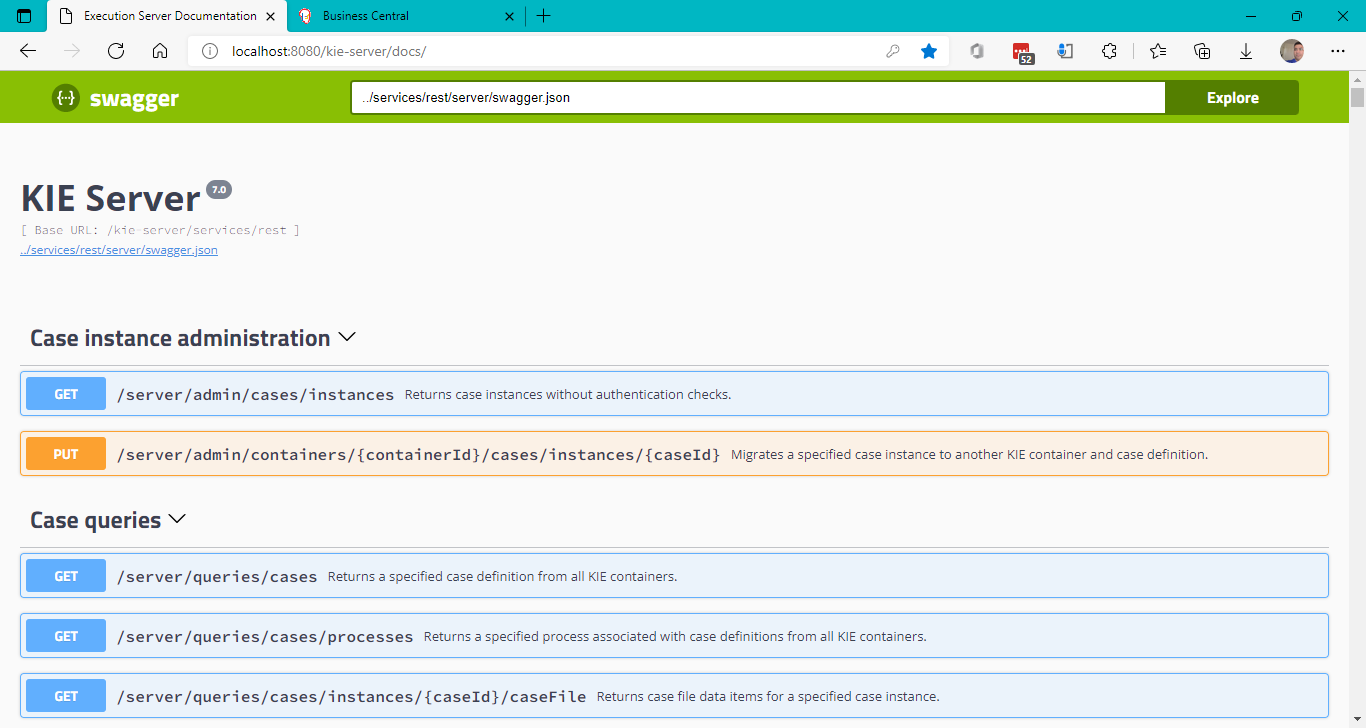


Figure jBPM KIE Server REST API Swagger doc

### Try examples

You can import some pre-built examples from <https://github.com/droolsjbpm/jbpm-playground>**.**

**Go to Spaces/My Space (or your own space) and click on Add project dropdown and click “import project” to import projects from the above GitHub uri.**

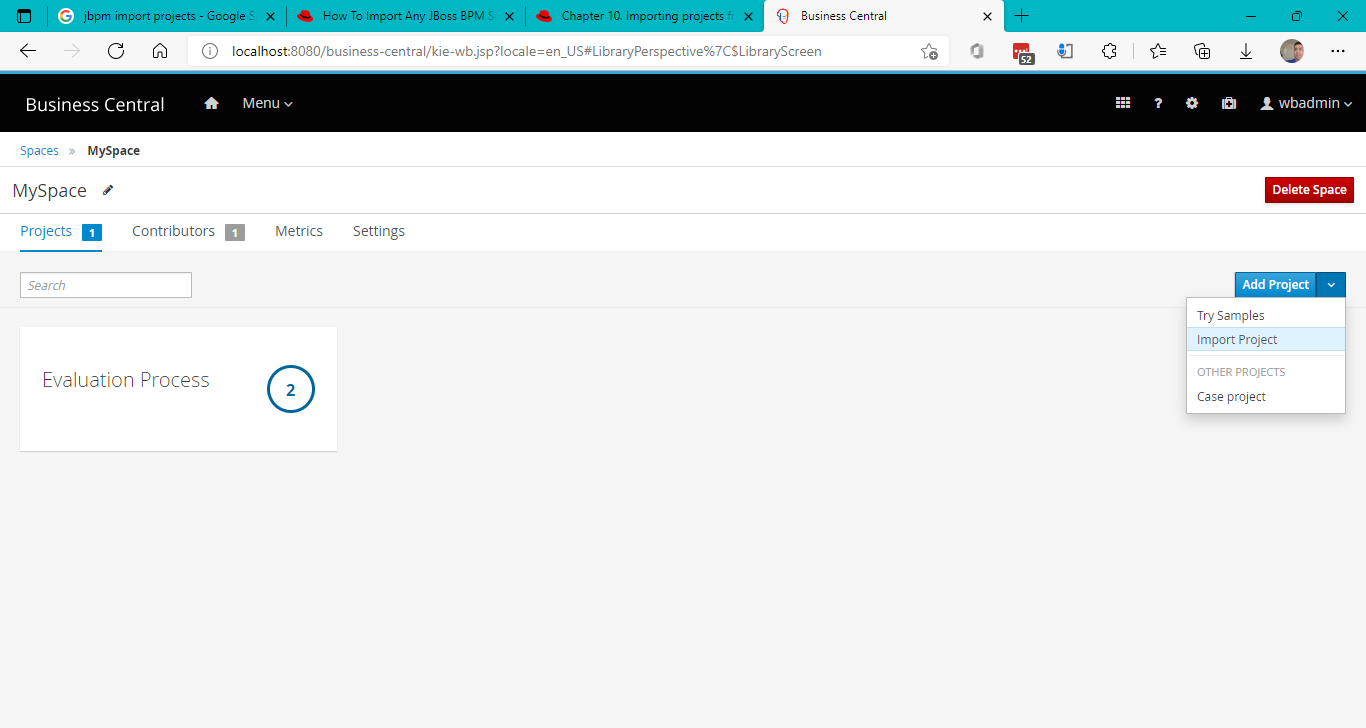


Figure 6 Import project from GitHub

Once you get yourself familiar with the tools of the jBPM, it’s time to see something running. The easiest way is to try one of the examples shipped with the platform, it will show typical path users take to design, build and execute business logic.

#### Evaluation process

|  |  |
| --- | --- |
| Evaluation process is a business process that is human centric (heavily uses human actors to perform work) that defines a complete flow of activities to perform employee evaluation. |  |

# ASP.NET Core web app which works with a sample jBPM workflow using jBPM REST API

## Web app architecture

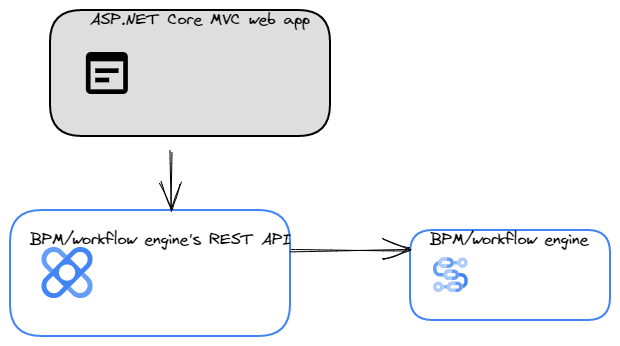


Figure ASP.NET Core MVC Web app architecture

## The Business Process/Workflow

The following is the process/workflow we will be coding using our web app UI app invoked the jBPM REST API to execute the whole process (all its human and system tasks) through our web app UI without even going to the jBPM management console.

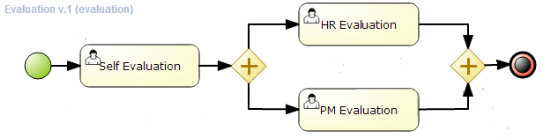


Figure HR Evaluation business process/workflow

## UI Web app

### Web Application Roles Mapped to BPM users and roles

You can create your own users in the ASP.NET Core web app with the database running on your local PostgreSQL instance and then edit the mapping of your users with the jBPM users and roles.

The below code snippet is an example of how Web app users are mapped to BPM users and roles.

public static class WebAppAndBPMUserMap

{

public static List<mapping> mappings { get; }

static WebAppAndBPMUserMap()

{

mappings = new List<mapping>();

mappings.Clear();

// These are example mappings. Please edit for your own

// env.

mappings.Add(new mapping

{

UserNameOnWebApp = "jack@example.com",

UserIdOnBPMEngineOrApp = "4",

UserRoleOnBPMEngineOrApp = "claim\_initiator",

UserNameOnBPMEngineOrApp = "walter.bates",

PasswordOnBPMEngineOrApp = "bpm",

UserIdOnBPMEngineOrApp2 = "",

UserRoleOnBPMEngineOrApp2 = "invoice\_initiator",

UserNameOnBPMEngineOrApp2 = "demo",

PasswordOnBPMEngineOrApp2 = "demo",

UserIdOnjBPMEngineOrApp = "",

UserRoleOnjBPMEngineOrApp = "employee",

UserNameOnjBPMEngineOrApp = "jack",

PasswordOnjBPMEngineOrApp = "jack",

});

mappings.Add(new mapping {

UserNameOnWebApp = "a@b.com",

UserIdOnBPMEngineOrApp = "4",

UserRoleOnBPMEngineOrApp = "claim\_initiator",

UserNameOnBPMEngineOrApp = "walter.bates",

PasswordOnBPMEngineOrApp = "bpm",

UserIdOnBPMEngineOrApp2 = "",

UserRoleOnBPMEngineOrApp2 = "invoice\_initiator",

UserNameOnBPMEngineOrApp2 = "demo",

PasswordOnBPMEngineOrApp2 = "demo",

UserIdOnjBPMEngineOrApp = "",

UserRoleOnjBPMEngineOrApp = "pm",

UserNameOnjBPMEngineOrApp = "john",

PasswordOnjBPMEngineOrApp = "john",

});

mappings.Add(new mapping

{

UserNameOnWebApp = "p@q.com",

UserIdOnBPMEngineOrApp = "15",

UserRoleOnBPMEngineOrApp = "claim\_reviewer",

UserNameOnBPMEngineOrApp = "mauro.zetticci",

PasswordOnBPMEngineOrApp = "bpm",

UserIdOnBPMEngineOrApp2 = "",

UserRoleOnBPMEngineOrApp2 = "invoice\_approver",

UserNameOnBPMEngineOrApp2 = "demo",

PasswordOnBPMEngineOrApp2 = "demo",

UserIdOnjBPMEngineOrApp = "",

UserRoleOnjBPMEngineOrApp = "hr\_admin",

UserNameOnjBPMEngineOrApp = "wbadmin",

PasswordOnjBPMEngineOrApp = "wbadmin",

});

}

}

#### PM ([a@b.com](mailto:a@b.com)) can start new employee evaluation

#### Employee ([jack@example.com](mailto:jack@example.com)) can self-evaluate themselves

#### HR ([p@q.com](mailto:p@q.com)) to evaluate an employee as HR

### Landing page of UI

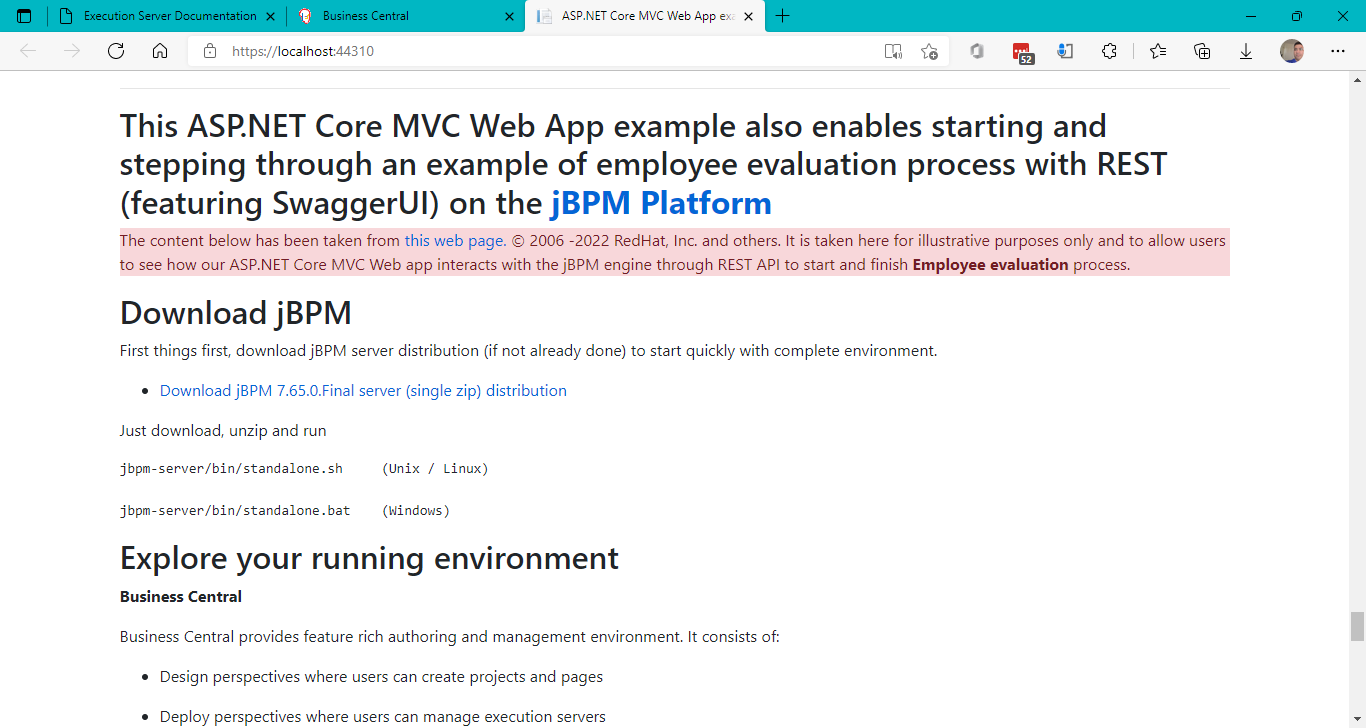


Figure ASP.NET Core MVC Web application

## Go to My Evaluations (sample pre-built HR evaluation workflow/business process in jBPM)

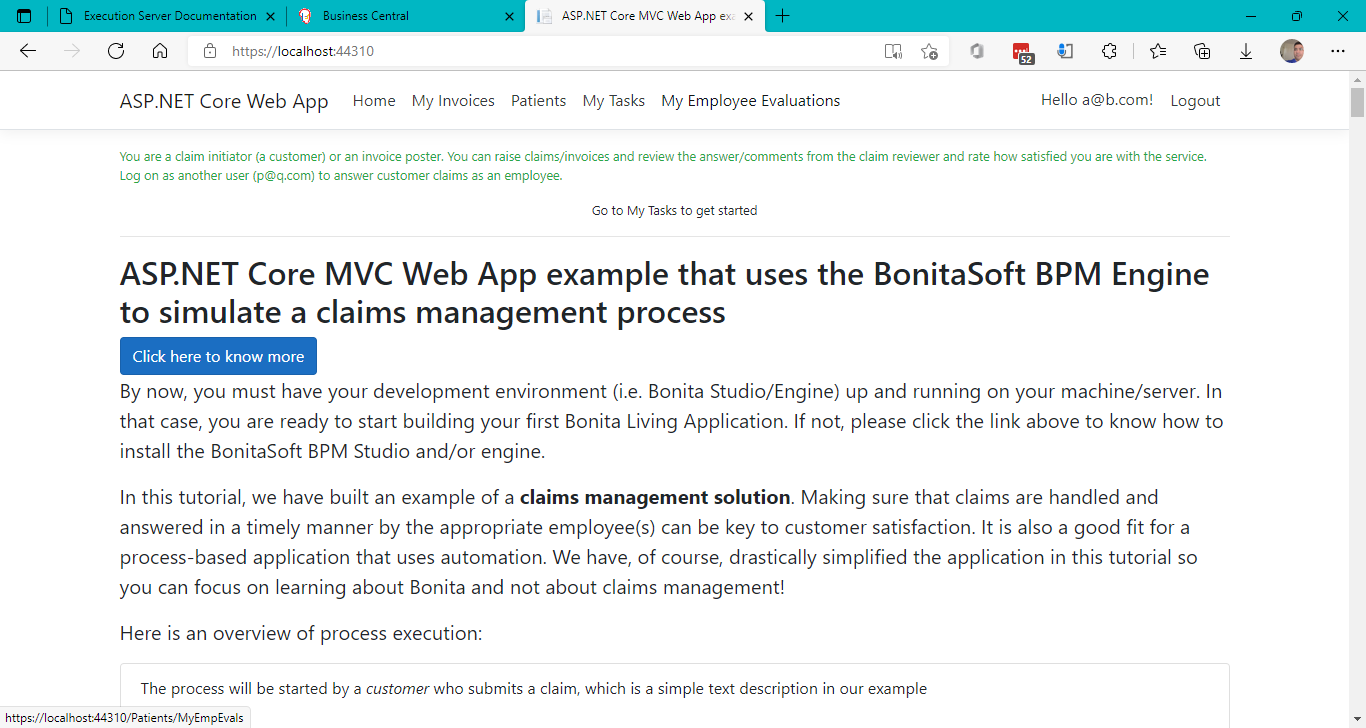


Figure Click on My Evaluations Link

### Start a new employee evaluation as a PM

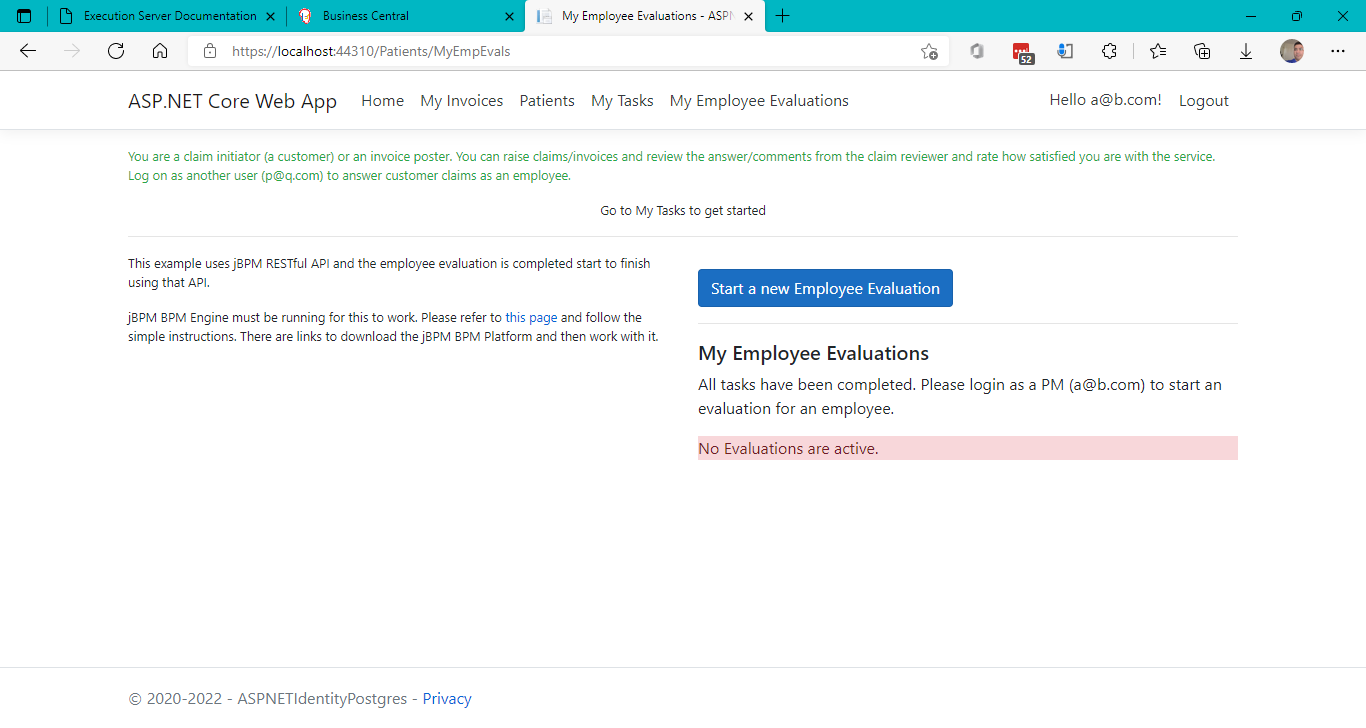


Figure Start a new Employee Evaluation as a PM

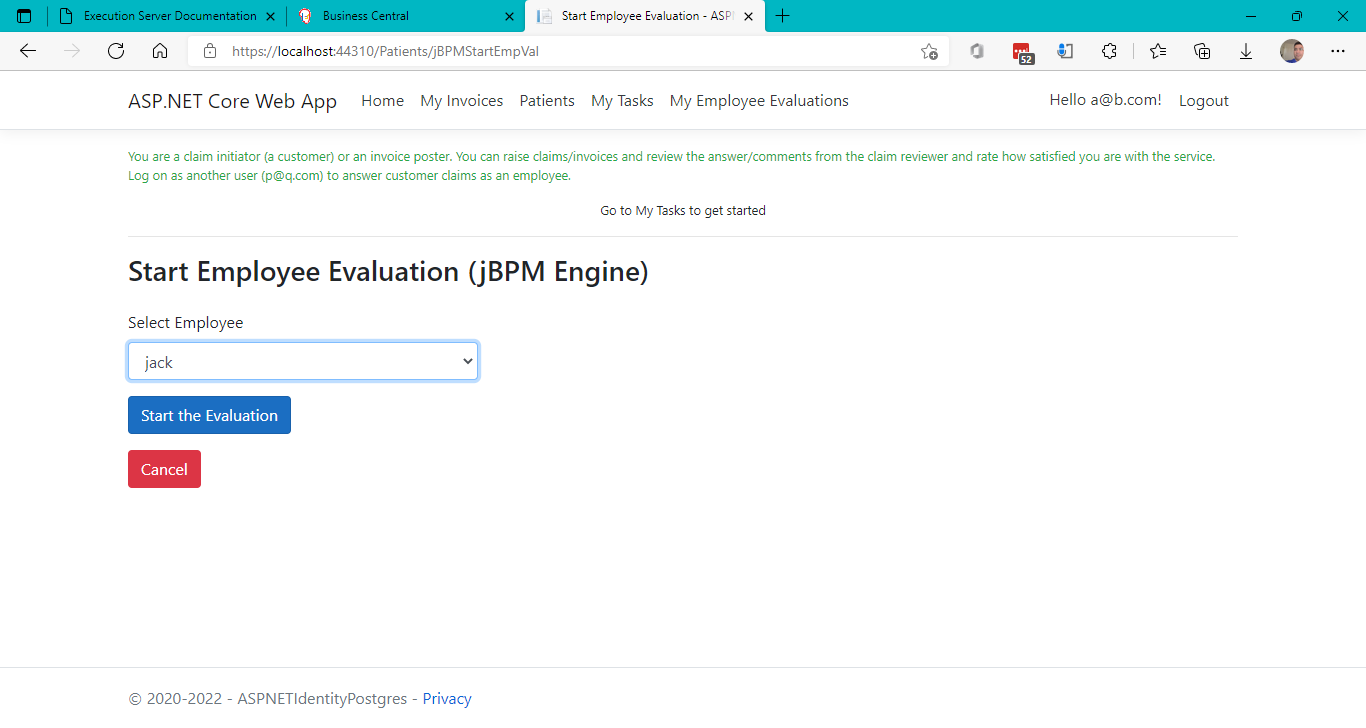


Figure Click the button "Start the evaluation"

The above action initiates a new jBPM process instance of the “evaluation” process.

See the image below.

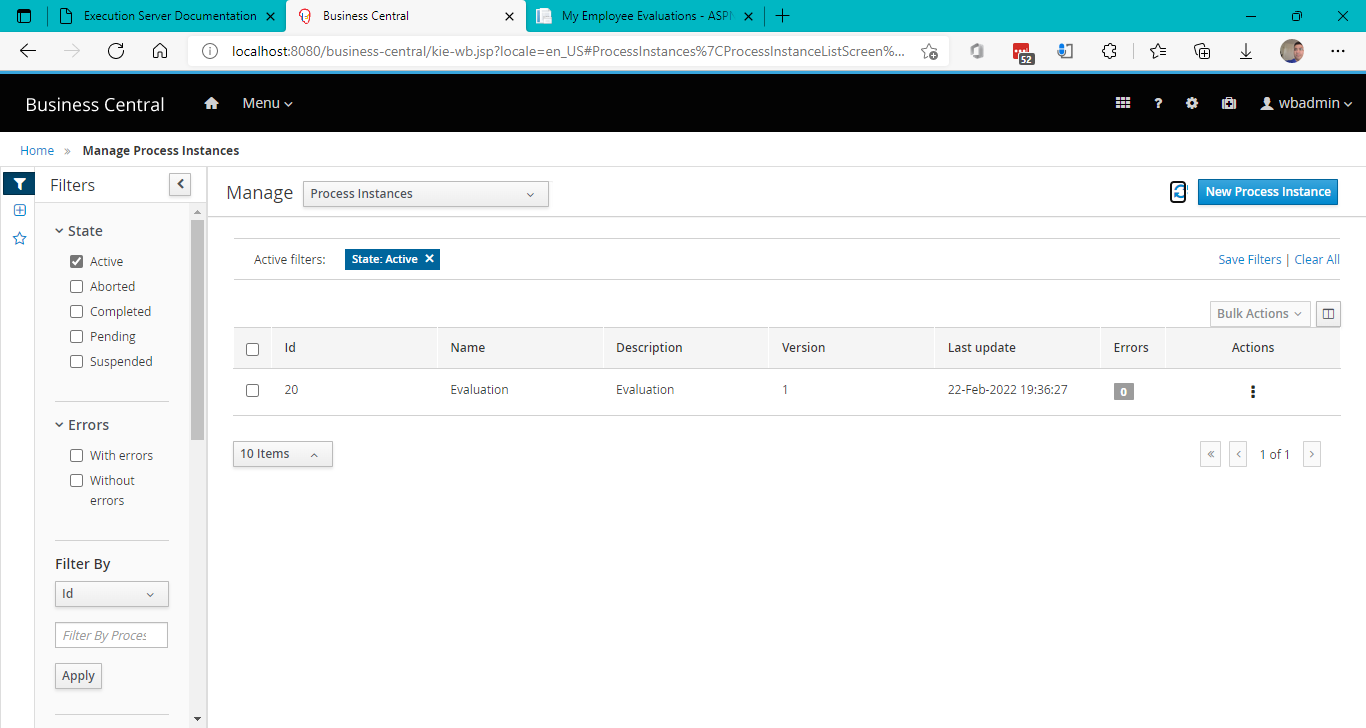


Figure New jBPM process instance triggered from the web app

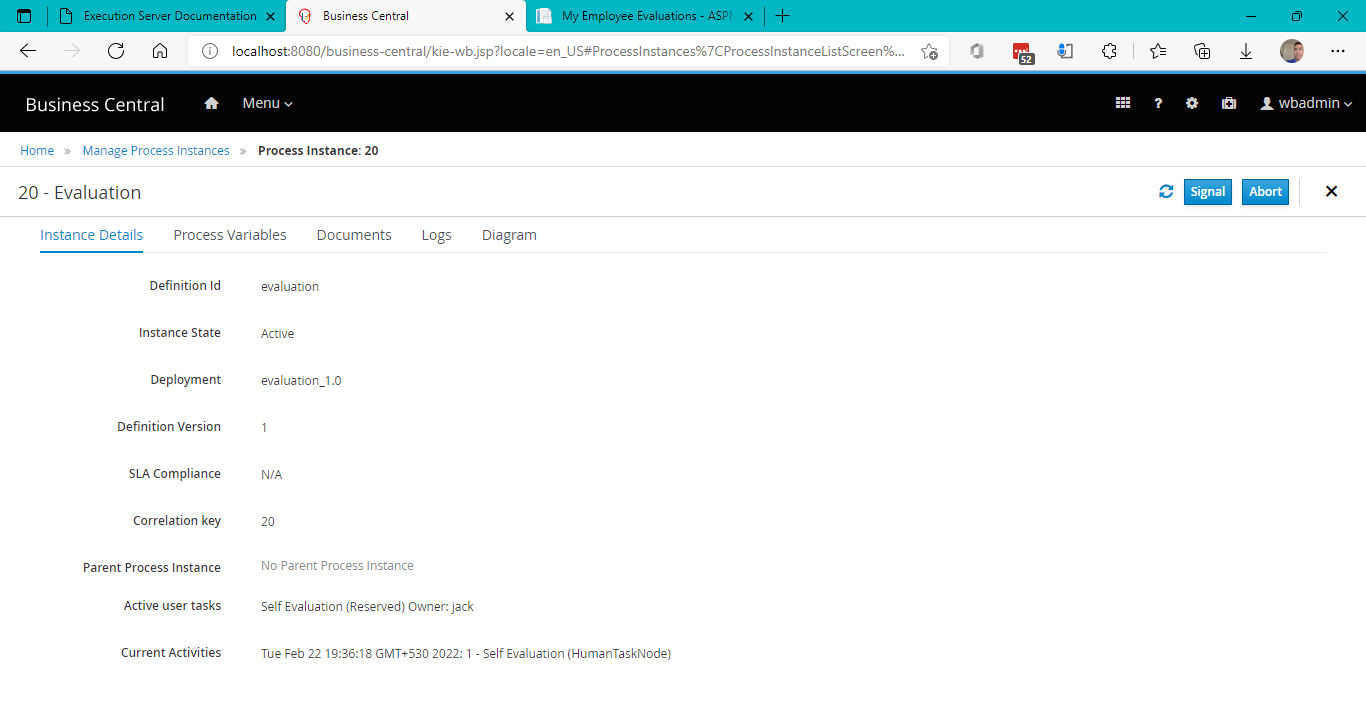


Figure New process instance status in jBPM console (business central)

### Login as [jack@example.com](mailto:jack@example.com) to evaluate yourself

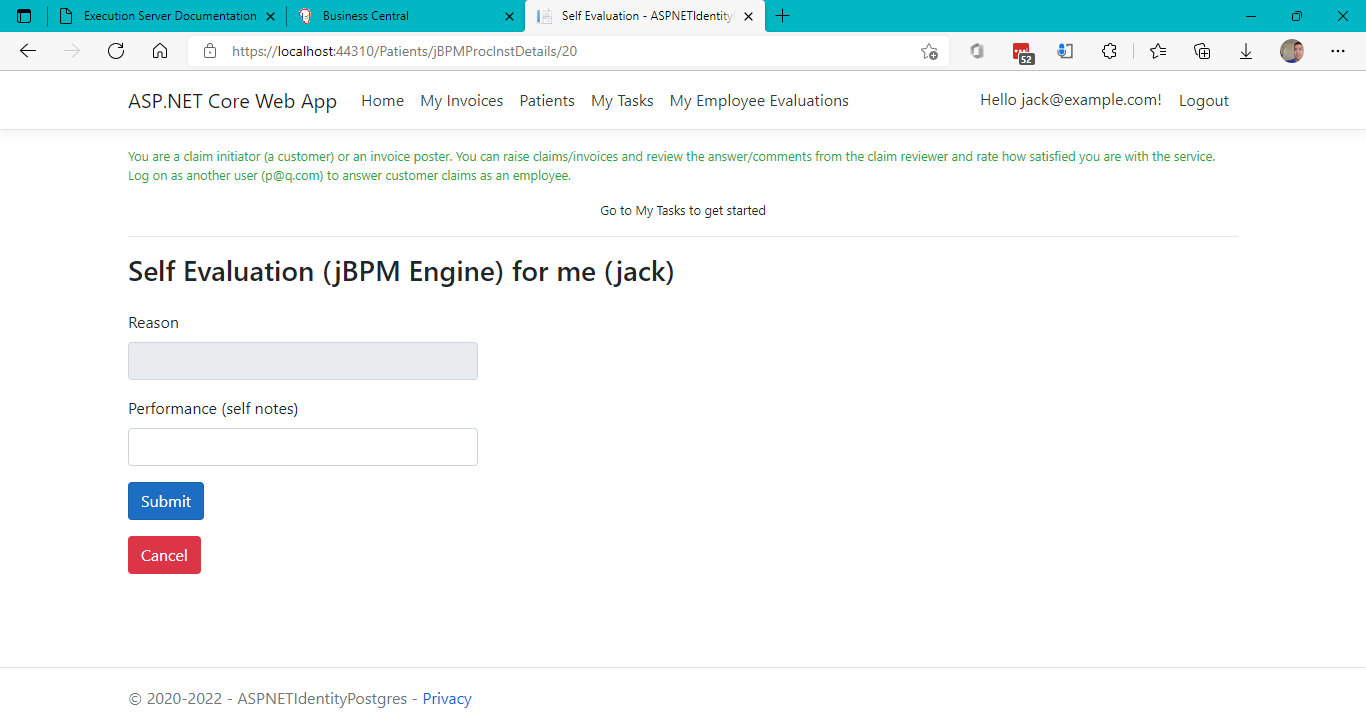


Figure Self evaluation

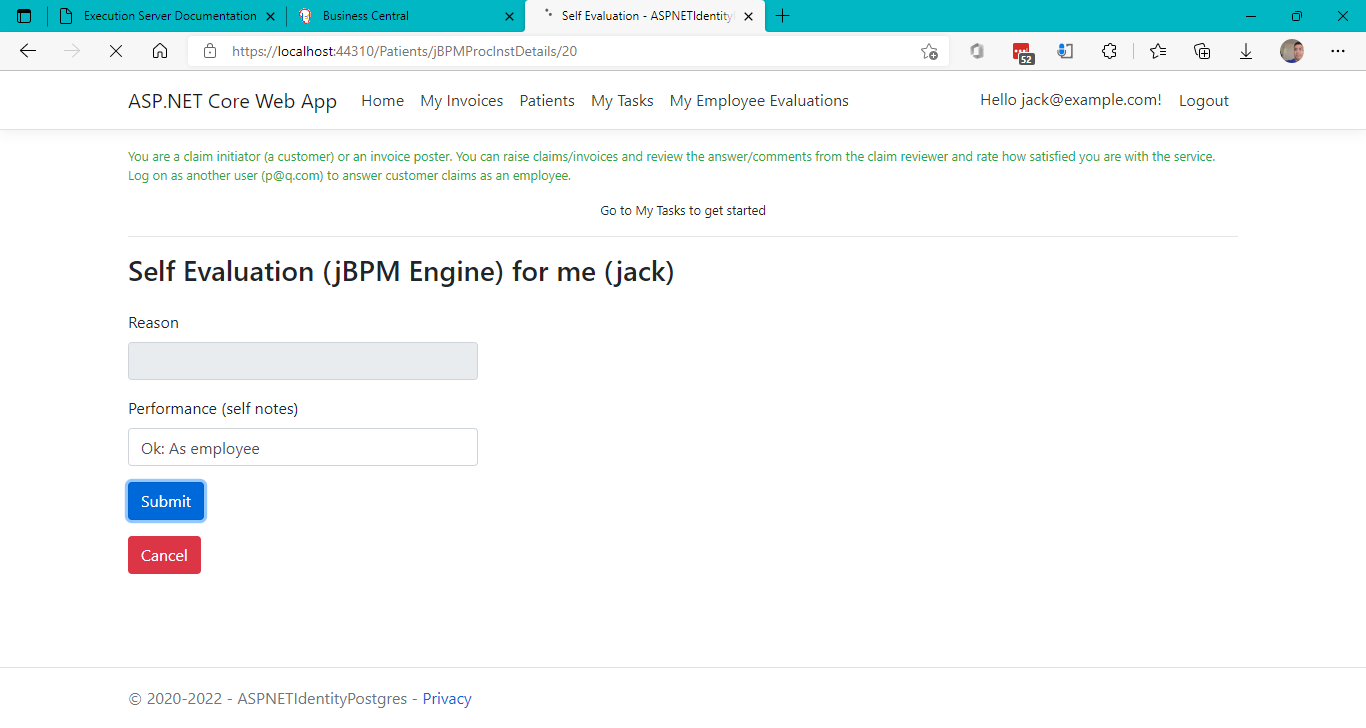


Figure Submitting self-evaluation

### Login as PM ([a@b.com](mailto:a@b.com)) again to do PM evaluation

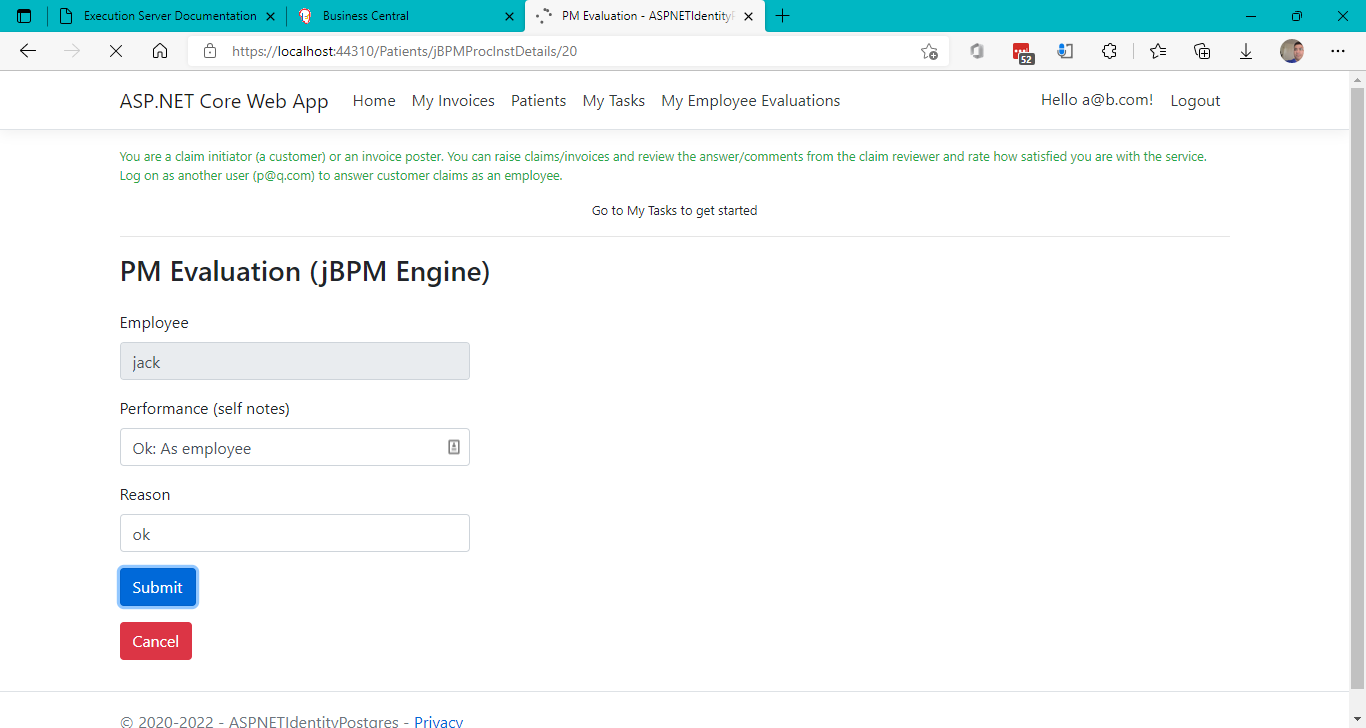


Figure Submitting PM Evaluation

If you click again on one of the tasks in the My Evaluations page, you will now see a message that the next step is the HR evaluation.

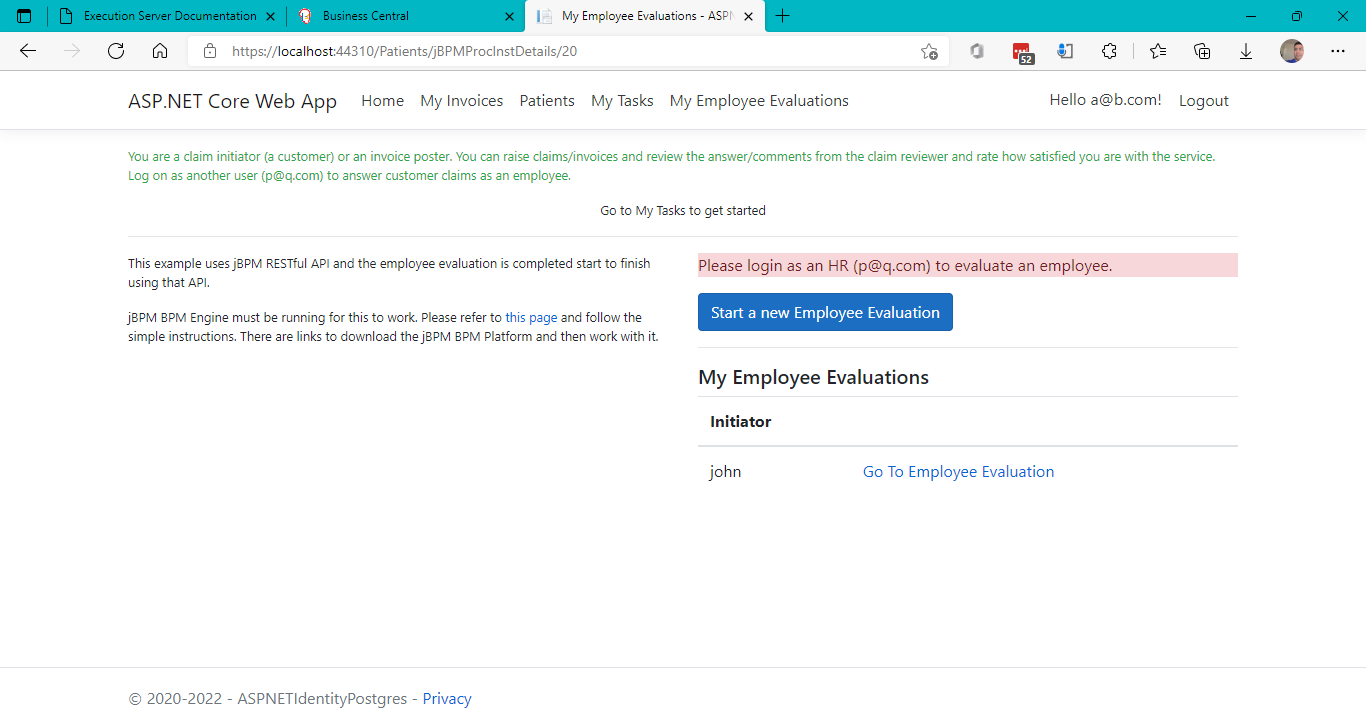


Figure Next step after PM evaluation is HR evaluation

### Login as HR ([p@q.com](mailto:p@q.com)) to evaluate an employee as HR

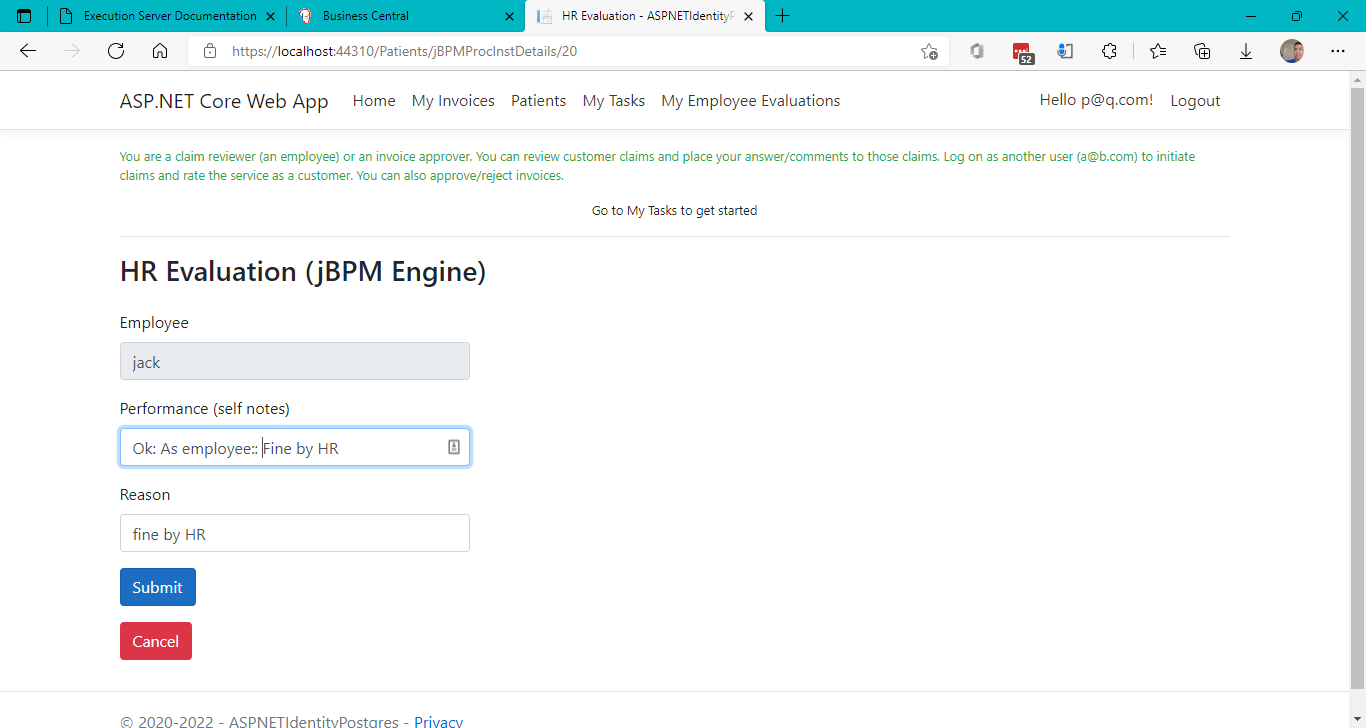


Figure Submitting HR evaluation

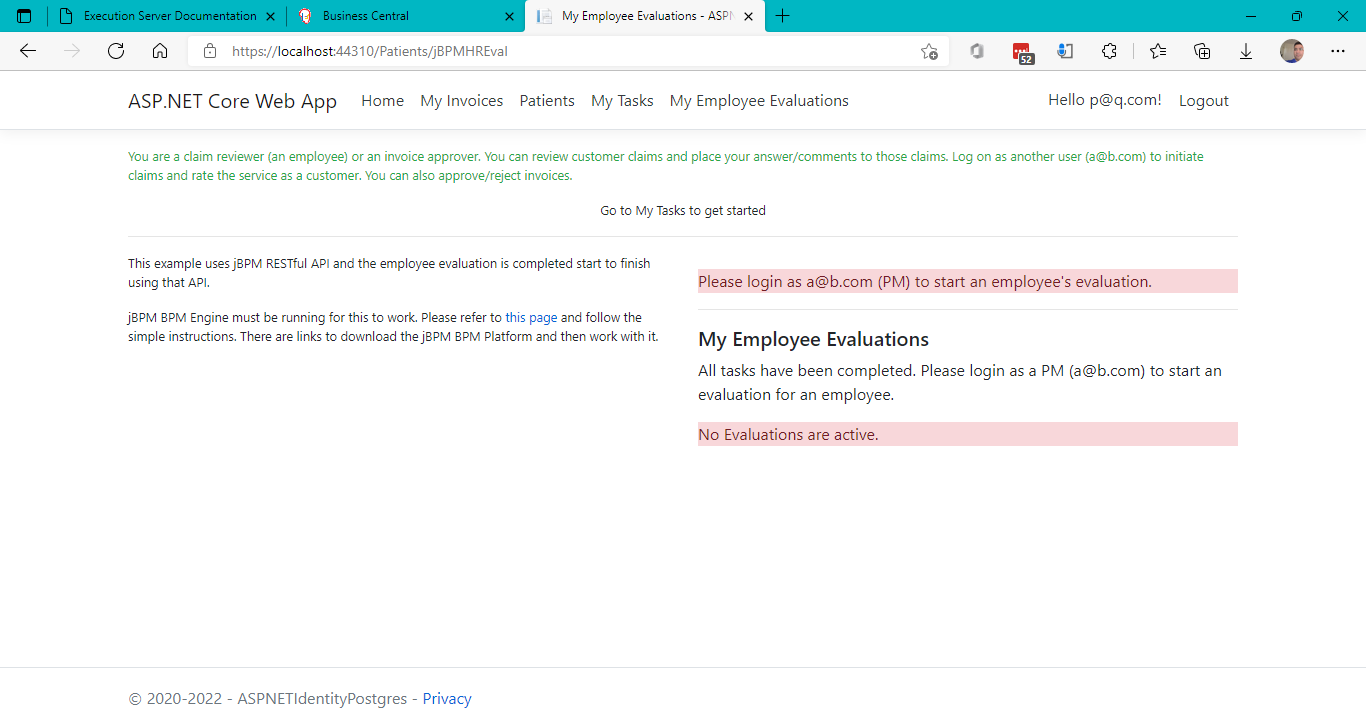


Figure After HR evaluation, the employee evaluation process/workflow is completed

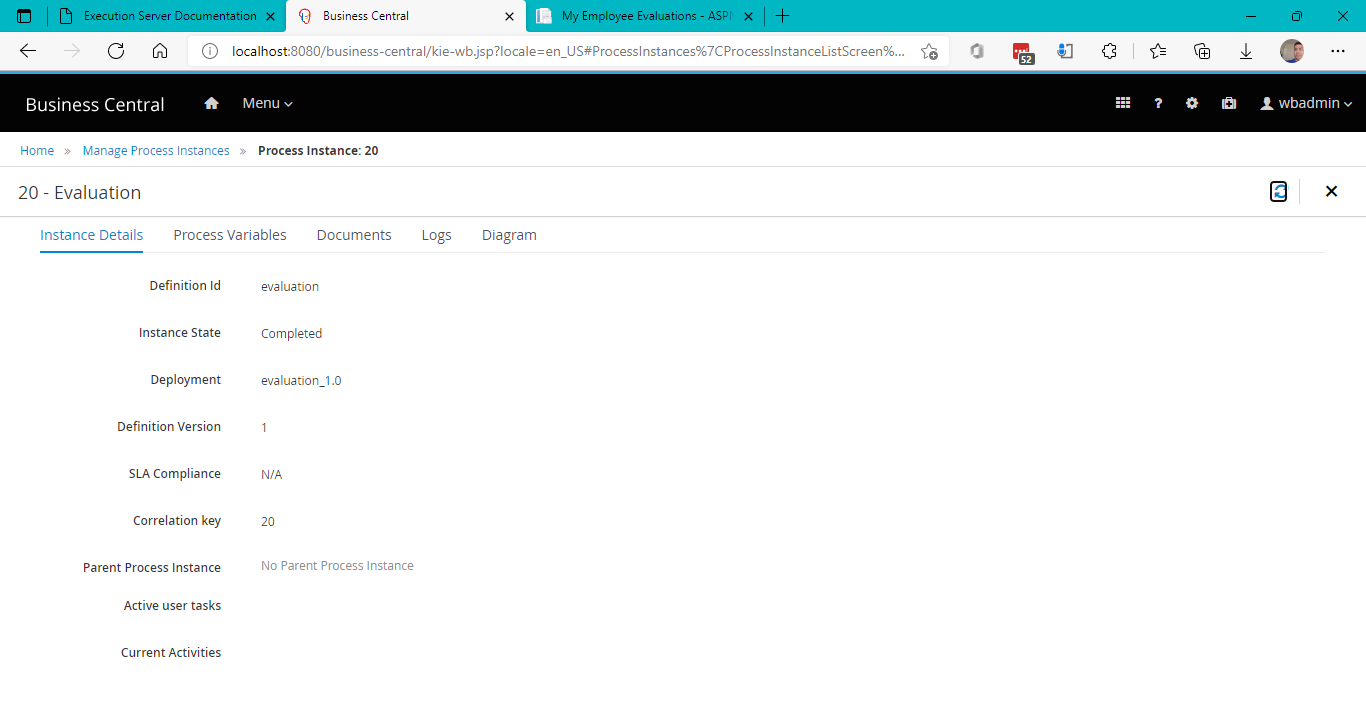


Figure New employee evaluation process completed status in jBPM business central

**Thus, we proved that the jBPM example process can be complete end-to-end using jBPM REST API (exposed via jBPM Kie Server REST instance) from the ASP.NET Core MVC Web application.**

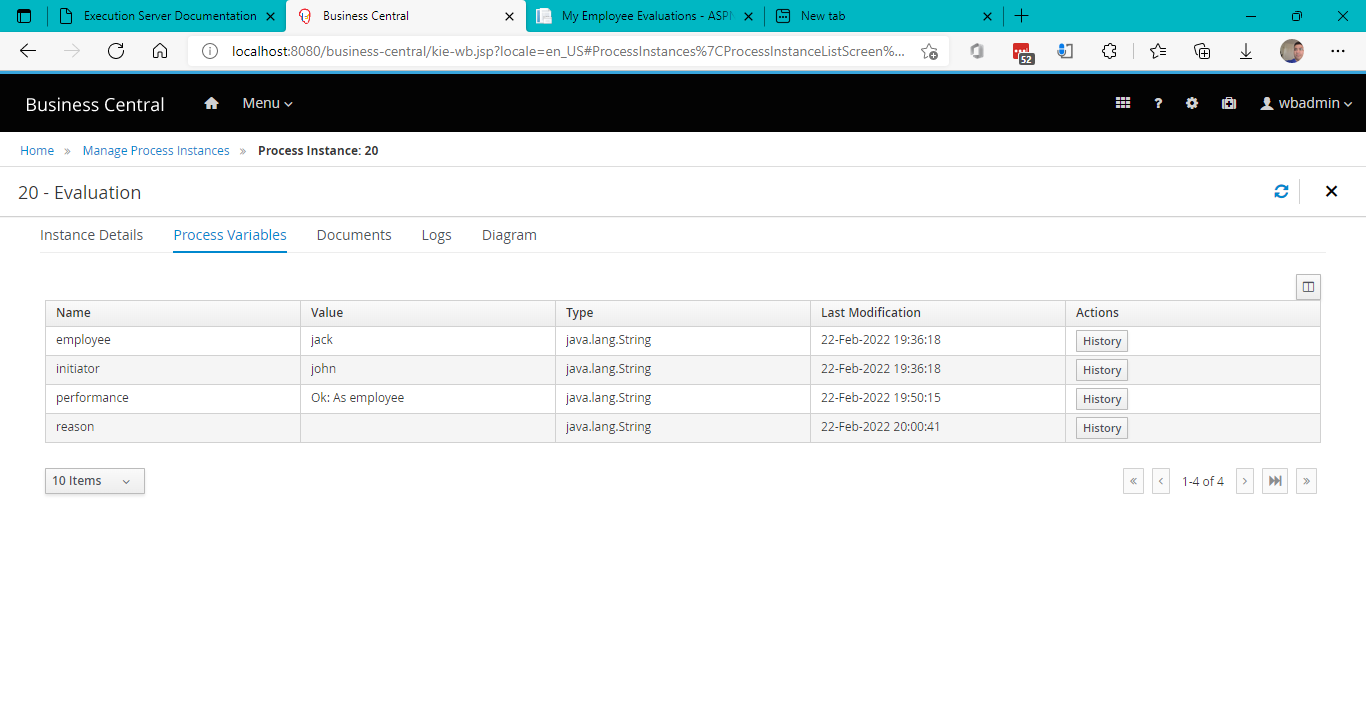


Figure Process instance variables set through UI. Since this is an example process, some variables might not get carried exactly. With a new process designed from the ground up, this can be more clear and visible.

# References

1. [jBPM - Open Source Business Automation Toolkit - Getting started](https://jbpm.org/learn/gettingStarted.html)