

PROCESS MINING PROJECT

Arcangeli Matteo

Giovanni Michetti

Francesco Feliziani

Nasa Rizoanun

A.Y. 2021/2022

Summary

[Introduction 3](#_Toc105942991)

[Tools and processes 3](#_Toc105942992)

[Model implementation 3](#_Toc105942993)

[Modell structure 4](#_Toc105942994)

[Customer 4](#_Toc105942995)

[Employee transfert 4](#_Toc105942996)

[End of employment 4](#_Toc105942997)

[External collaboration 4](#_Toc105942998)

[New employee 4](#_Toc105942999)

[Suppliers 5](#_Toc105943000)

[Conclusion 5](#_Toc105943001)

# Introduction

Main target of the project is to demonstrate the advantages of a BPMN Engine in the enterprise to carry on and track most of the repetitive tasks and processes happening in the organization.

The scope of this section of the project is

* Model the actual interactions (as described in this document) between the involved groups of people in the “User Life Cycle Process” in a BPMN 2.0 format, including information moving between involved groups of people.
* Propose one or more versions of the model with some automation included, meaning with some Manual Task converted to Service Tasks, explaining how the process should change, if needed, and how the automated task should be implemented.
* Optionally providing one or more implementations of some of the service tasks identified by the optimization step.

# Tools and processes

BPMN diagrams and implementation should be produced using the Camunda platform, comprising the Camunda Engine Community Edition and the Camunda Modeler.

Examples of the messages exchanged between the systems will be provided, and all enterprise systems interactions, will be represented in the form of c# interfaces, that can be either implemented on a test system or entirely mocked.

# Model implementation

Initially we designed the models following Loccioni’s requirements, we decided to implement six different diagrams in different file instead of one containing all the information; we took this decision to better manage the information and the code; in fact, the code of each diagram in easier and better understandable.

In each model the service task has been implemented as external task, we took this decision because we use a mock code, just to simulate the token flow inside the model; so, in the future it will be possible replace each implementation of external task with real implementation of the service. The struct of this program is composed of a main function and different method, each of this uses *ExternalTaskClient*, as we said before these functions are only to simulate the behaviour, so each task wrote a simple message into the log, if it is necessary, we print the value of the needed variables, each client subscribe a specific topic.

In each diagram we implemented the message exchange using Java class inside the model, we decided to correlate each variable into the model using the appropriate java method; each message correlation had been implemented using a different java class and a different message id.

During the model design we decide to automatize some tasks that was previously manually executed.

# Modell structure

Each model is composed of two folders one containing the model project and one containing the external worker project; after build the model project we put the generated *.war* file into the webapp folder of tomcat.

# Models Details

In the following chapter we are going to explain in more details each model.

## Customer

Customer process start getting all information related to him, after that a ticket is open to IT department, and customer information are stored into User Management System, following a role is assigned to the customer, an email with the information is sent one to MGR department for ticker closing and another mail is sent to the customer. It is important to underline the different implementation of the message sent in fact because of customer is an empty pool the message sent has been implemented has external task to avoid execution error (we apply this approach in all the other diagram in which we have empty pools), finally the ticker is close even in the IT department.

## Employee transfert

Initially in HR department get information about employee id and the new role are asked, then the new data are store into HR system, following an email is sent to IT department, this will generate the new profile, after these two emails are sent one to HR department for closing ticker and one to the employee, even in this case the email to employee has been implemented as external task.

## End of employment

Initially we asked for the ID of employee that he wants to end his employment, then the data are store in HR system, after those two emails are sent the former to it department that will deactivate the user in Loccioni’s system; the other is sent to Administration department to manage the paperwork.

## External collaboration

Initially the MGR department ask for external collaborator information, then a ticker is created to IT department and an email with needed information are sent to this one. The IT department verify the information, IT department evaluate if it is a new collaboration or an extension of an old one, to simulate this step, this task has been implemented as Java class, this class evaluate if the employee ID is 1 simulate that is an existing collaboration, otherwise it means that it is new collaboration. In case of existing collaboration, the end date is update; otherwise, a new use user and a profile are created, then the credential are notified to the external employee and the ticket is closed.

## New employee

The HR department collect information about new employee after that a new entry is created in HR system, following an email is sent to IT department that will create a new entry in User Management system and new profile will be created, then an email with new credential is sent to the employee.

## Suppliers

In the MGR department add information about new suppliers and store it into ERP system, following a ticker is open and an email is sent to IT department that will create a new entry into intranet application and a new role is asked and the ticker is close, following an email is sent to MGR department that will notify the supplier with an email.

# Conclusion

Thank to this project we made practice of Camunda into a real case provided by Loccioni group, we made practice of BPMN modelling and implementations using Java.