Business Process Automation Lab Demonstrator 2

(Guided Project)

User manual

in Winter 2023/24

TH Köln - Campus Gummersbach

Faculty of Computer Science and Engineering Science (F10)

Project by: Rahib Nazir Butt (11155035)

Berrak Kücük (11160144)

Project Supervisor: Prof. Dr. Matthias Zapp

Date: 1st of February, 2024

Technology
Arts Sciences
TH Köln

Contents

Introduction

Business Process Automation Lab Demonstrator 2 is a guided project offered in Winter 2023/2024 semester. The project consists of an end-to-end process of a small bicycle manufacturing factory. From customer order to order management, production to shipment, the whole process consists of several different technologies implemented all together into one solution.

Intended use

This business process automation lab demonstrator can be implemented and used inside a small bicycle factory that designs and manufactures different types of bicycles like Mountain Bicycle, Hybrid Bicycle and Electric Bicycle. This demonstrator has the functionality of handling different processes from different departments inside the factory like customer order management, production of the bicycle along with the shipment, movement of products inside the warehouse and IoT devices to monitor the whole warehouse.

About the Camunda engine

For this project, the complete solution is implemented using Camunda 8 self-managed version using non-production environments. Using the basic components of Camunda platform, all the required components as Docker containers the project is executed.

For more information, you can access the official Camunda platform link here: https://github.com/camunda/camunda-platform

Requirements

To run Camunda 8 self-managed, your system should meet the following minimum hardware requirements:

Minimum hardware requirements:

- Processor: Dual-Core or Quad-Core CPU, 2 GHz or faster
- RAM: 2 GB (4 GB recommended)
- Disk Space: At least 1 GB free disk space for installation and application server

Note regarding requirements:

It's important to note that these are the minimum requirements, and depending on the size of your processes, the number of users, docker containers and other factors, you may need to allocate more resources to your system to ensure smooth and efficient operation.

User interface overview

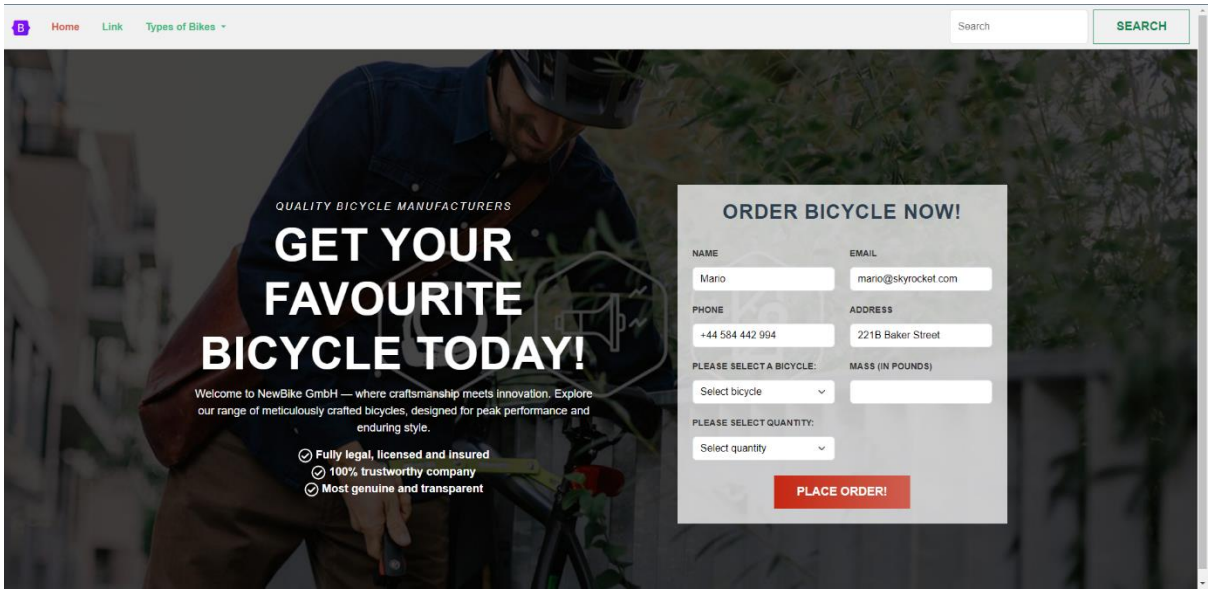

Figure 1 Single page front-end application

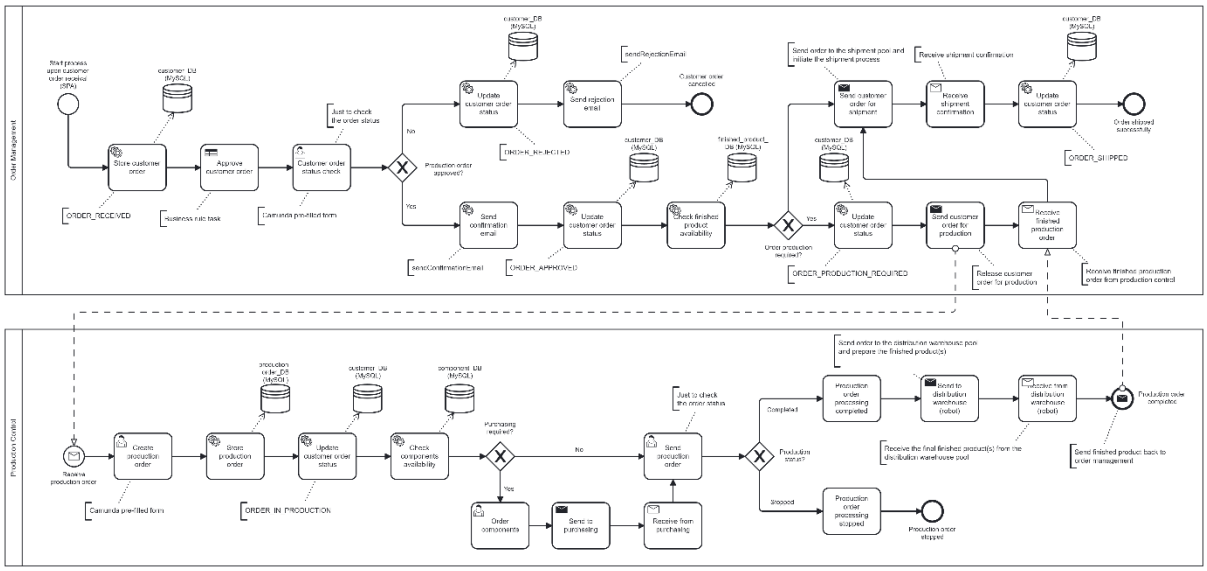

Figure 2 BPM diagram overview

Features

- Single command to download all the libraries/dependencies
- Single command to run the project including all the containers
- Single command to remove all the containers
- Monolithic architecture to connect user interface with Zeebe
- Microservice architecture for all the job workers and different processes
- Complete Dockerized solution
- Tracking of process instance state
- Front-end customer order application (responsive)
- Business rule tasks (DMN)
- Customer order status emails
- MySQL database to track the customer order
- User tasks
- Service tasks/job workers
- User assignments
- Send/receive tasks
- Message start/stop events
- Message throw events
- Expressions
- Exclusive gateways
- Timer events
- Connectors
- Warehouse operations (Fischertechnik robots)

Prerequisites

- Docker Desktop Application (mandatory)

Testings

This project has been successfully tested on three different machines. Two of them were based on Windows system and the third using macOS.

Git versioning

The project is completely versioned using Github and can be shared amongst the interested user.

Get in touch

If you have any questions or concerns regarding the project, please feel free to write down an email at my address: rahibbutt@gmail.com and I will be really delighted to help you out with all your queries!

----------THE END----------