Low code? nah…
but less code? Yeah!

Maciej Swiderski

OpenEnterprise

who am I?

- independent software engineer and consultant
- workflow enthusiast in the field for more than 15 years
- creator of Automatiko project
- write blog posts to share knowledge
- occasionally tweets about workflows and software development

@SwiderskiMaciek

Low code

Targets less technical users to deliver IT solutions quickly

- Use graphical user interface to assemble working software
- Create solutions by using predefined set of features
- Usually constraint to given domain (e.g. marketing campaigns)

- Not as appealing to developers as regular “development work”

Less code

Allows developers to be more productive by focusing on business logic

Is just that...

less code

Less and less code...

Spring and hibernate xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:context="http://www.springframework.org/schema/context"
  xsi:schemaLocation="http://www.springframework.org/schema/beans
  http://www.springframework.org/schema/beans/spring-beans.xsd
  http://www.springframework.org/schema/context/
  http://www.springframework.org/schema/context/spring-context.xsd">

  <bean id="operations"  class="com.acme.spring.beans.Operations"></bean>
  <bean id="employee"  class="com.acme.spring.beans.Employee"></bean>
  <bean id="department"  class="com.acme.spring.beans.Department"></bean>
</beans>
```

Less and less code...

Annotations

```
1  @Entity
2  public class Vehicle {
3
4      @Persistent
5      protected String vehicleName = null;
6
7      @Getter
8      public String getVehicleName() {
9          return this.vehicleName;
10     }
11
12     public void setVehicleName(@Optional vehicleName) {
13         this.vehicleName = vehicleName;
14     }
15 }
```

Less and less code...

Auto configuration

```
1  @Configuration
2  @EnableAutoConfiguration(exclude={DataSourceAutoConfiguration.class})
3  public class MyConfiguration {
4  }
5
6
7  @Service
8  public class DatabaseAccountService implements AccountService {
9
10   private final RiskAssessor riskAssessor;
11
12   @Autowired
13   public DatabaseAccountService(RiskAssessor riskAssessor) {
14     this.riskAssessor = riskAssessor;
15   }
16 }
```

Less and less code...

Wiring

```
1
2  public class Receiver {
3
4    @Incoming("my-channel")
5    public void consumePayload(Price payload) {
6        // do something
7    }
8
9  }
```

We got used to less code already, right?

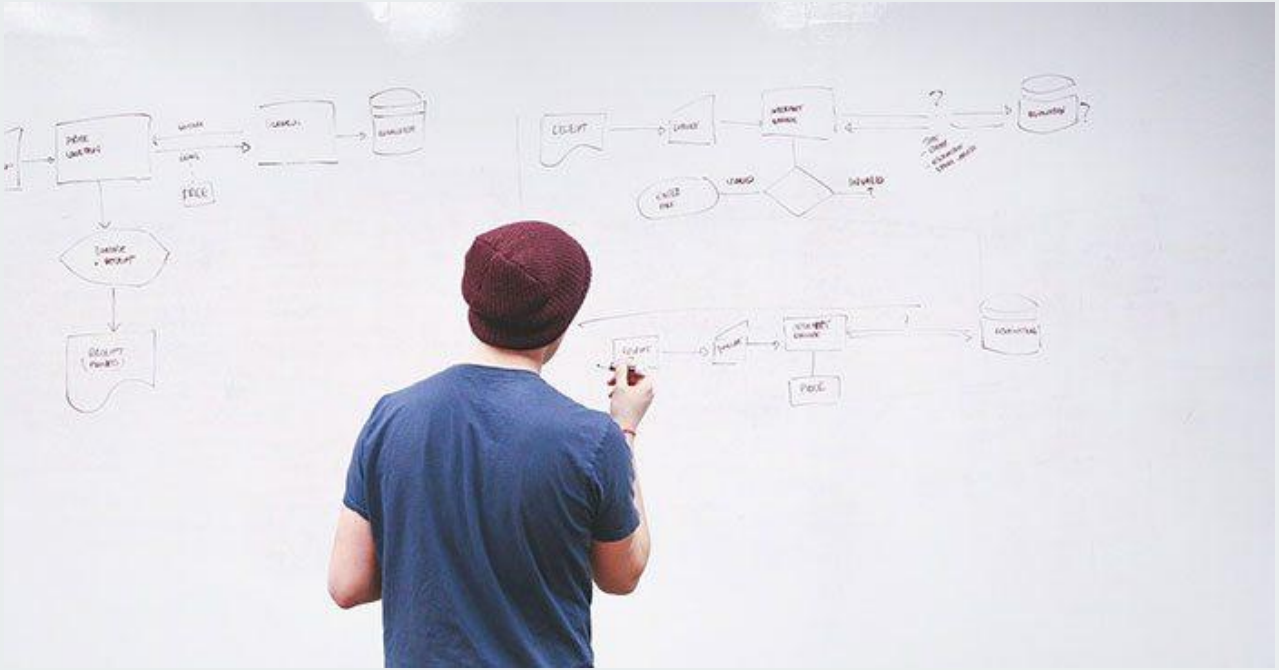
Workflows

Workflows are not just for orchestration

- Everything we do in software industry is a process
- Workflows allow to simplify communication between team members
- Workflow automation comes with handy features that would have to be written
- Comes with isolation by design - workflow definition vs workflow instance

Why workflows?

We all start with ...



... why not to take it further?

source https://officeninjas.com

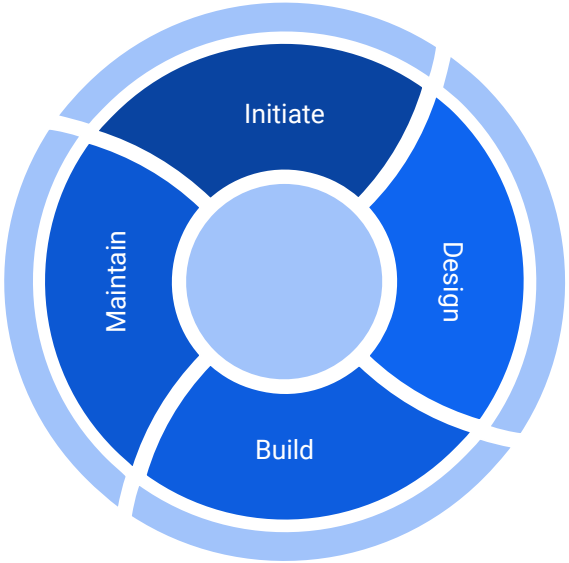
Automatiko

- Automatiko - framework to build services and functions based on workflows
  - workflow as a service
  - workflow as a function
  - workflow as a function flow
- Covers quite few use cases
  - Event streams (e.g. based on Apache Kafka, IoT based on MQTT)
  - Business entity life cycle and database record processing
  - Service orchestration
  - Kubernetes Operators
  - Human centric or batch processing
- Open source - Apache license
- Built on top of Quarkus and Microprofile

Building parts service

manage life cycle of parts

inspired automotive industry



Requirements

- Service needs to handle life cycle of the part - from initial creation up to retirement
- Parts are uniquely identified by part number
- Parts can be accessed at any time by their part number
- Part has well defined data model
- Parts go through number of phases
  - design,
  - development,
  - testing,
  - production
- Part information is of interest for other (downstream) services

Let’s build it!

@automatiko_io

Wrapping up

Less code means more efficient software deliveries

- Less code allows developers to be more productive
- Less code does not mean there is no code, it is just written by someone else ;)
- Workflows can improve visibility of the business logic
- Less code and workflows are enablers to build low code platforms

Want to know more?

- Get in touch on twitter
  - **@SwiderskiMaciek**
  - **@automatiko_io**

- Visit website
  - https://automatiko.io
  - https://github.com/mswiderski/devoxxpl-2022

If you are interested and would like to explore more don't hesitate to reach out!

Questions?