TagItSmart
Smart tags driven service platform for enabling ecosystems of connected objects

Grant agreement 688061

SO Engine
Experimentation report

Deliverable ID:	**D3.3**
Deliverable Title:	Experimentation report
Project acronym	**SO Engine**
Project title:	**Smart Offers Engine for retail**
Extension acronym:	
Extension title:	
Contractual date of delivery:	**5/1/2018**

Contents

Introduction ***3***

SO Engine platform ***3***

1.1 Web platform **3**

1.2 SO Engine mobile application **6**

Scenarios ***12***

Conclusion ***13***

Figures

Figure 1 - Web platform 3
Figure 2 - Rule creation 4
Figure 3 - Thing creation 5
Figure 4 - List all things 5
Figure 5 - Tag linked to a thing (generated by TIS platform) 6

Introduction

We describe here the web platform and the mobile application, and some scenarios inspired by TagItSmart use cases.

SO Engine platform

1.1 Web platform

A test platform for SO Engine is available at http://185.161.45.105:5000/.

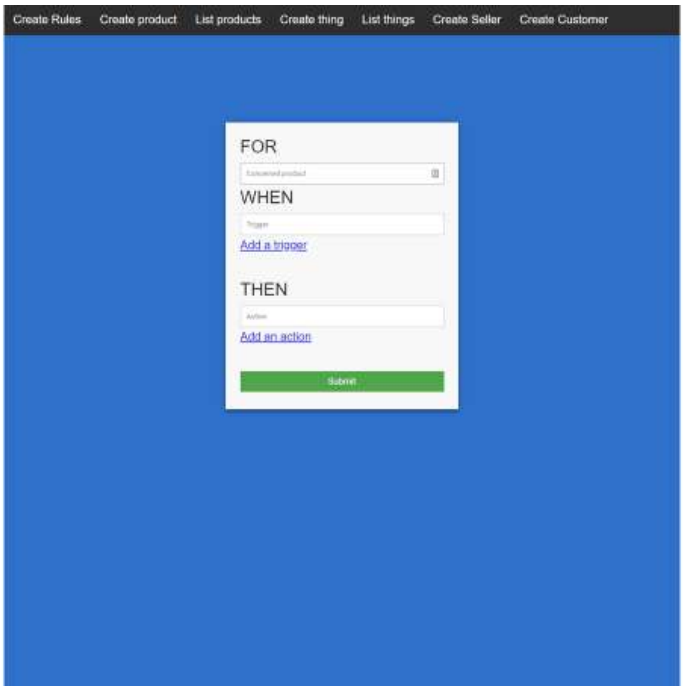

Figure 1 - Web platform

The platform allows to create very simply FOR-WHEN-THEN rules

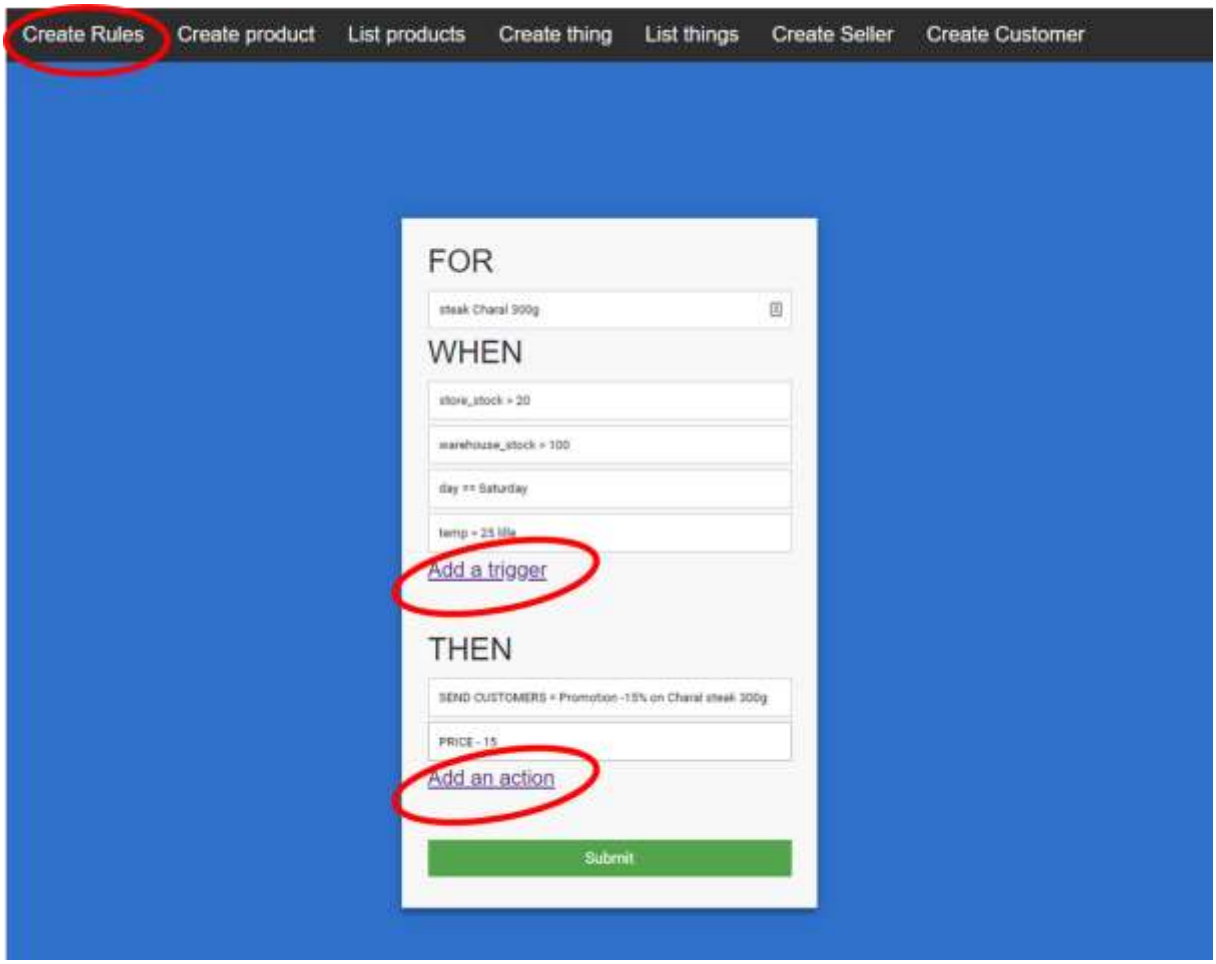

Figure 2 - Rule creation

It is possible to add a product, a thing, a customer, a seller, to list all the products of the bottom and all the things

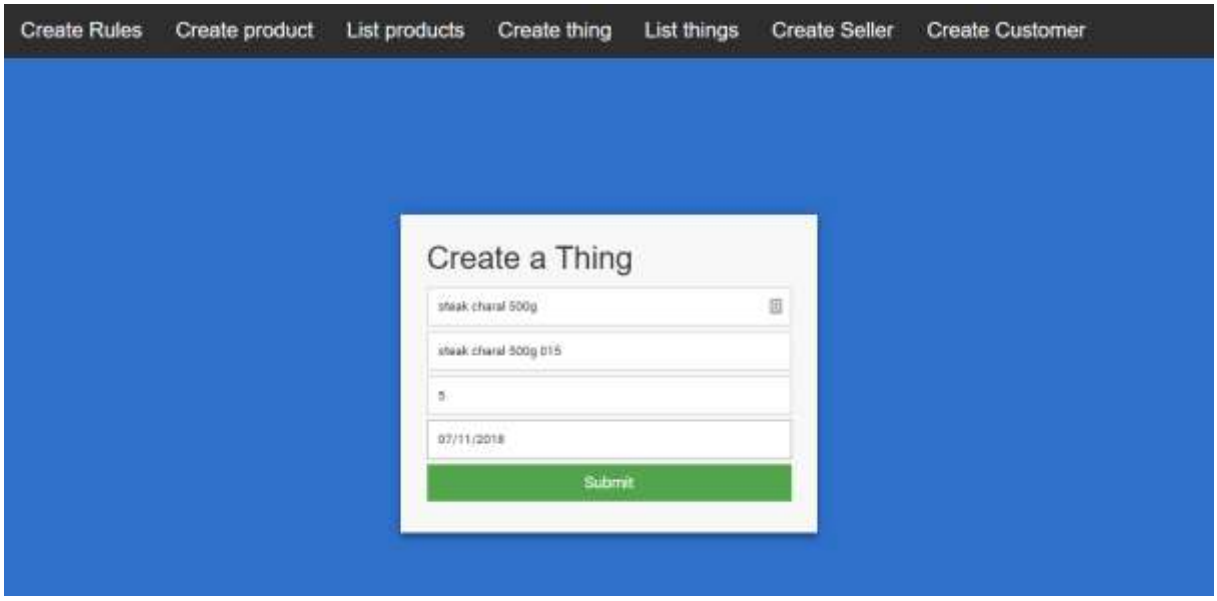

Figure 3 - Thing creation

Create Rules | Create product | List products | Create thing | List things | Create Seller | Create Customer

Product Name	Thing Name	Price	Date of deployment	Expiration Date	Location	ID	Short ID	QR code URL
steak charal 500g	steak charal 500g 001	5.0	no	07/20/2018	warehouse	UpegxQtmBXPa9Kwwa2GpPCks	cgK6k3Li	QR Code url
steak charal 500g	steak charal 500g 002	5.0	no	07/20/2018	warehouse	U5BXbttnBXPa9pawwFysMrXd	vtN7Ya8J	QR Code url
steak charal 500g	steak charal 500g 003	5.0	no	07/20/2018	warehouse	UKVgxQt5eXPwt5wwwkP4raYq	AaspFD1i	QR Code url
steak charal 500g	steak charal 500g 004	5.0	no	07/20/2018	warehouse	UKVDxQQMBD8wtKwRa3mhrMrh	IKM8BtP3	QR Code url
steak charal 500g	steak charal 500g 005	5.0	no	07/20/2018	warehouse	UpBXxttrMG8EhqaaRkpbPDmq	5xZaZm1u	QR Code url
steak charal 500g	steak charal 500g 006	5.0	no	07/20/2018	warehouse	U5BDb9tPeXswtKRaRFs4rcCe	zexiSwzG	QR Code url
steak charal 500g	steak charal 500g 007	5.0	no	07/20/2018	warehouse	U5eDbt9QegsRtKwwR3qaYrqt	fAFNABCg	QR Code url
steak charal 500g	steak charal 500g 008	5.0	no	07/20/2018	warehouse	UpVDcaabBD8al5wwRFsH7Anm	wpX9EZaR	QR Code url
steak	steak							QR

Figure 4 - List all things

Each Thing is linked to a tag generated by TIS platform.

Figure 5 - Tag linked to a thing (generated by TIS platform)

Actions resulting from the triggering of rules (such as price changes) are immediately reflected in the databases.

1.2 SO Engine mobile application

A mobile app has been developed (the android apk is available at http://185.161.45.105:5000/smart_offers.apk).

Functions for seller :

- modify the location of a Thing (warehouse, store, …) by scanning the tag
- read notifications (messages on promotions, …)
- delete notifications

Functions for consumer :

- read notifications

Figure 6 - SO Engine mobile application

1.3 Demo

A demo video is available at https://goo.gl/vPiKg1.

In this demonstration of the platform:

- The seller connects to the mobile application with his login and password (Figure 7)
- Seller chooses an action on a Thing (put on the shelf in store) (Figure 8)
- He scans the tag on a Thing (Figure 9)
- Stock is decreased in the warehouse and increased in the store (Figure 10)
- Seller creates a new rule that triggers a price reduction and promotion on beers for Thursday and a stock threshold(Figure 11)
- The price is updated on the rule trigger (Figure 12)
- The customer receives a promotion notification on his mobile application (Figure 13, Figure 14)

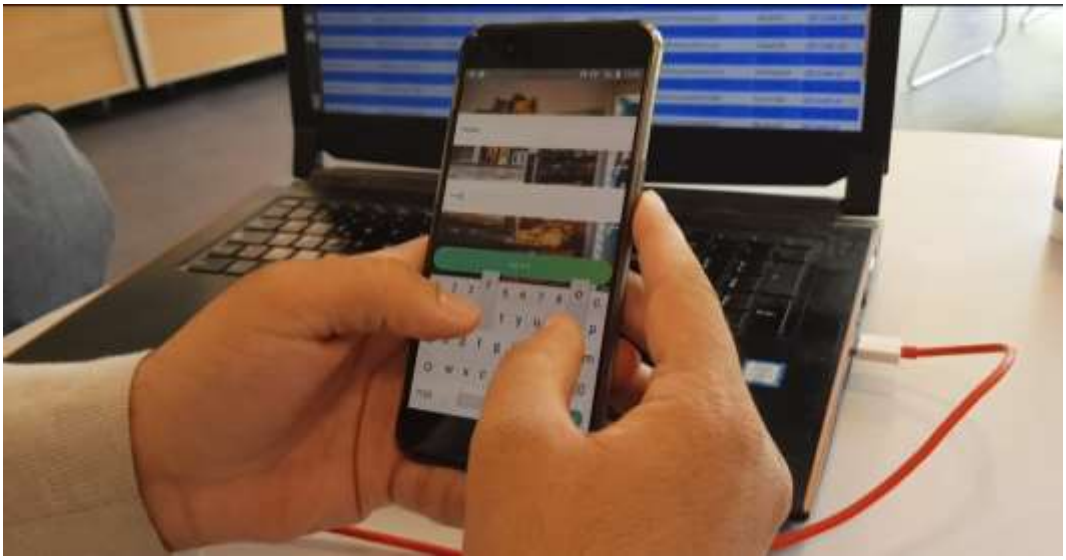

Figure 7 - Identification as seller

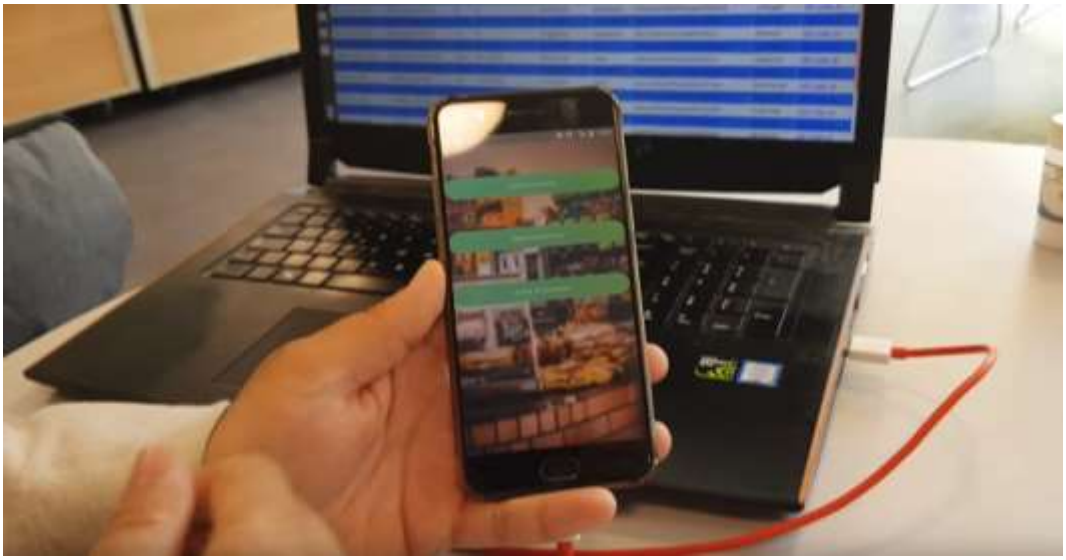

Figure 8 - Select the Action on a thing

Figure 9 - Scan the tag and move from warehouse to store

Figure 10 - The warehouse stock is updated

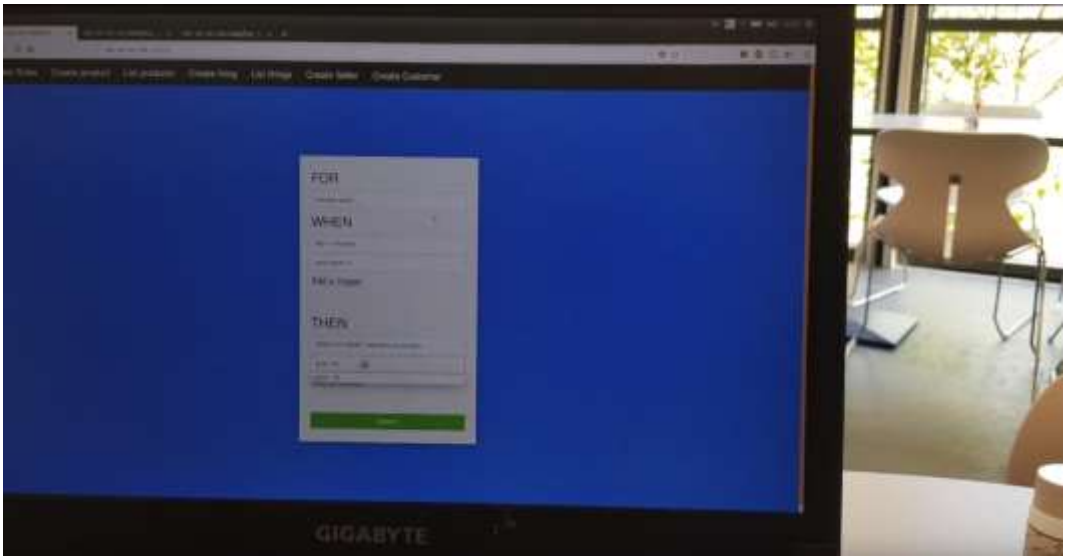

Figure 11 - Add a rule

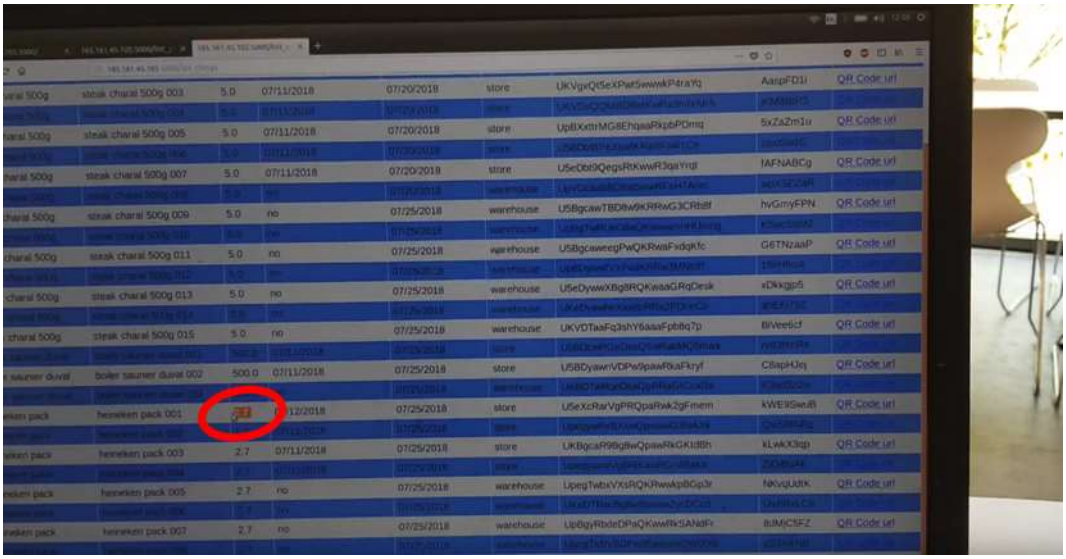

Figure 12 - The price is updated

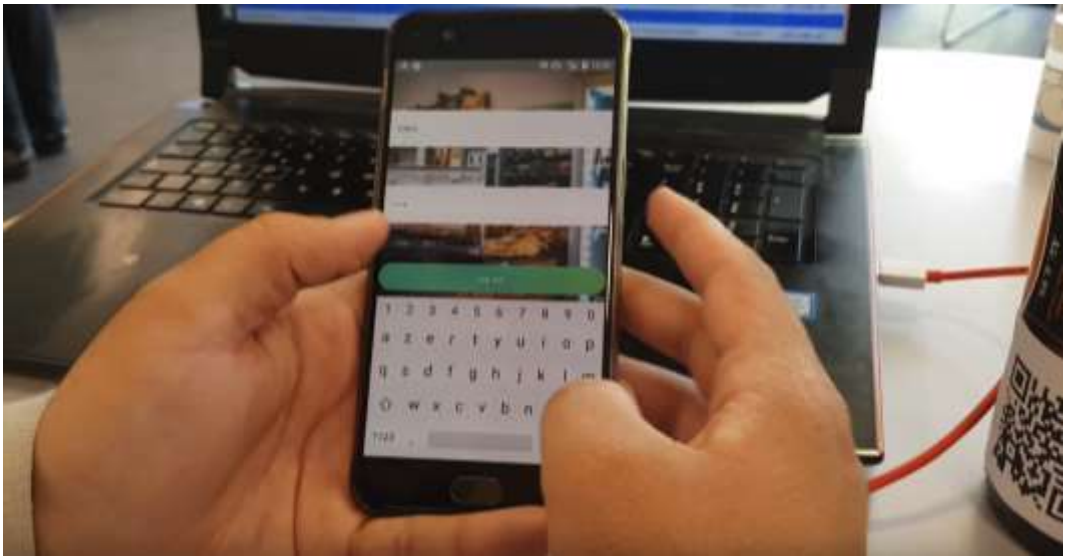

Figure 13 - Connect as customer

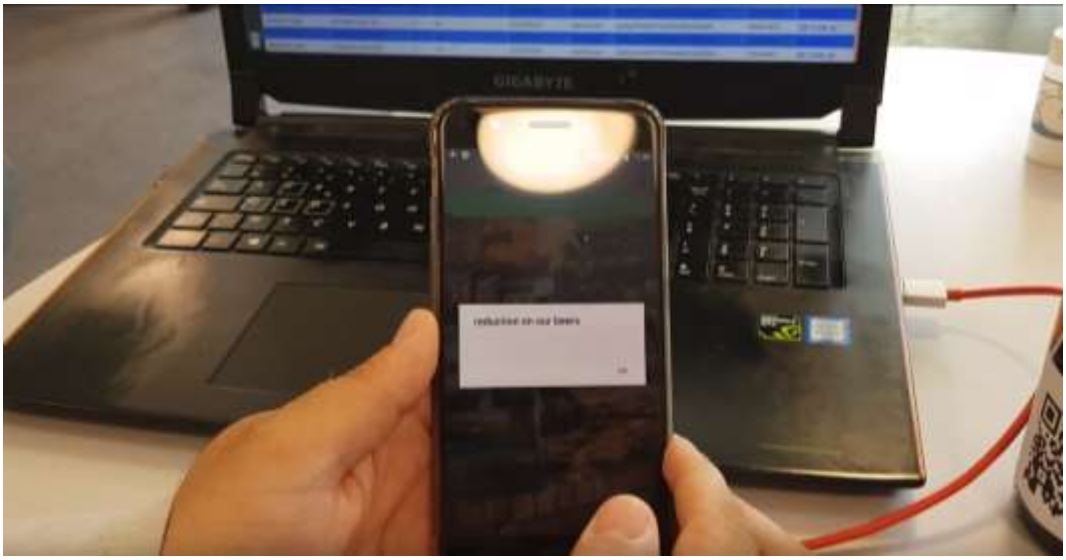

Figure 14 - A notification appears on the customer application

Scenarios

As we will explain in the impact report, it was not possible to convince a retailer to launch a full-scale experiment in the relatively short time frame of the project.

We have therefore developed some scenarios inspired by TagItSmart use cases.

For dynamic pricing, a first rule apply on products. For a store whose shelf refill is Monday, a promotion on meat of -15% will be triggered on Saturdays if the weather is hot (outside temperature above 25°C) and if stocks on the shelf and in the warehouse are still high. As soon as the conditions are all true, the rule is triggered :

```
FOR         Charal steak 300g
WHEN        store_stock > 20
            warehouse_stock > 100
            day == Saturday
            temp > 25 lille
THEN        SEND CUSTOMERS = Promotion -15% on Charal steak 300g
            PRICE - 15
```

Another example rule apply on Things. If there's a lot of steaks left on the shelf, and if the weather is fine, The price of all steaks with an upcoming deadline (less than three days) is reduced by 20% :

```
FOR         Charal steak 300g
WHEN        store_stock > 20
            expiration_date < 2
            weather == clear
```

```
THEN        PRICE - 20
```

For beers, if the weather is fine and temperature is high, and if it remains a high stock in the warehouse, then sellers will receive a message on the mobile application to launch a promotional operation

```
FOR         Heineken pack 20
WHEN        weather == clear
            temp > 25 lille
            warehouse_stock > 70
THEN        SEND SELLERS = Launch commercial operation on Heineken
```

For the Home Services use case, a very simple rule allows to send a message to the customer to indicate that maintenance needs to be scheduled after one year :

```
FOR         Saunier Duval boiler Thema
WHEN        today == release_date+365
THEN        SEND CUSTOMERS = Think about overhauling your boiler !
```

Conclusion

As we have seen, the platform would allow to **quickly develop new use cases** for TagItSmart. The abstraction layer allows to write rules without any programming skills. SO Engine is designed to allow **the retailer himself** to add new rules according to his marketing strategy.