



PROCESS and SERVICE DESIGN PROJECT

a.a. 2021-2022

SmartPharma

Aida Gasanova - 10732229

Matteo Makovec - 10782774

Xiaoyu Luo - 10777140





TABLE OF CONTENTS



01 Context

02 Process
description

03 Archimate model

04 Choreography
diagram

05 Orchestration
diagram

06 Executable
diagram

07 Petri Net
validation

08 Swagger API
documentation

09 Demo



01 CONTEXT



TOPIC:

Healthcare



**SOCIAL
CHALLENGE:**

Allows everyone to get
the needed medicines



MISSION:

Simplify the medicine
order process



SERVICE:

Automate and simplify
the process of
providing medicines

02 PROCESS DESCRIPTION 1/3

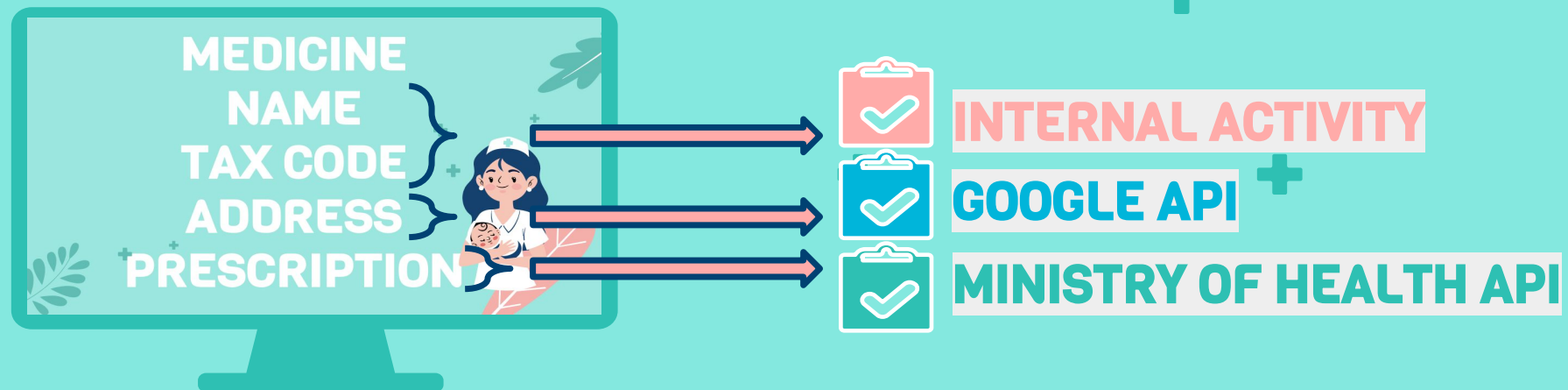
SmartPharma service allows to automate and simplify the process of providing medicine. The service is provided from a pharmacy to a common customer. It simplifies the process of ordering any medicine from the pharmacy.



When a customer wants to order a certain medicine, she/he will connect through a Web Application to submit a new order request. Once the request was submitted, the information provided by the customer is automatically checked. The information should include personal data (name, tax code), the delivery address, the name of the medicine and, optionally, the prescription.

02 PROCESS DESCRIPTION 2/3

The personal data consistency is automatically checked by SmartPharma. The delivery address is checked by contacting google api and receiving back the map's information. The prescription is also checked automatically by requesting a confirmation from the ministry of health.



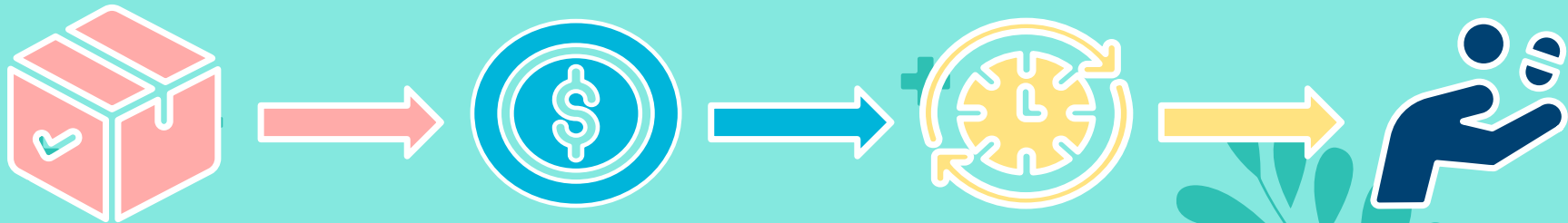
The provided information will be either approved or denied. In case of approval the requested medicine is automatically checked for availability in stock by SmartPharma. In case of denial or stock unavailability a notification is sent to the customer with the denial reason.

02 PROCESS DESCRIPTION 3/3

In case of stock availability, SmartPharma delegates a third-party payment system to finalize the purchase with the customer.

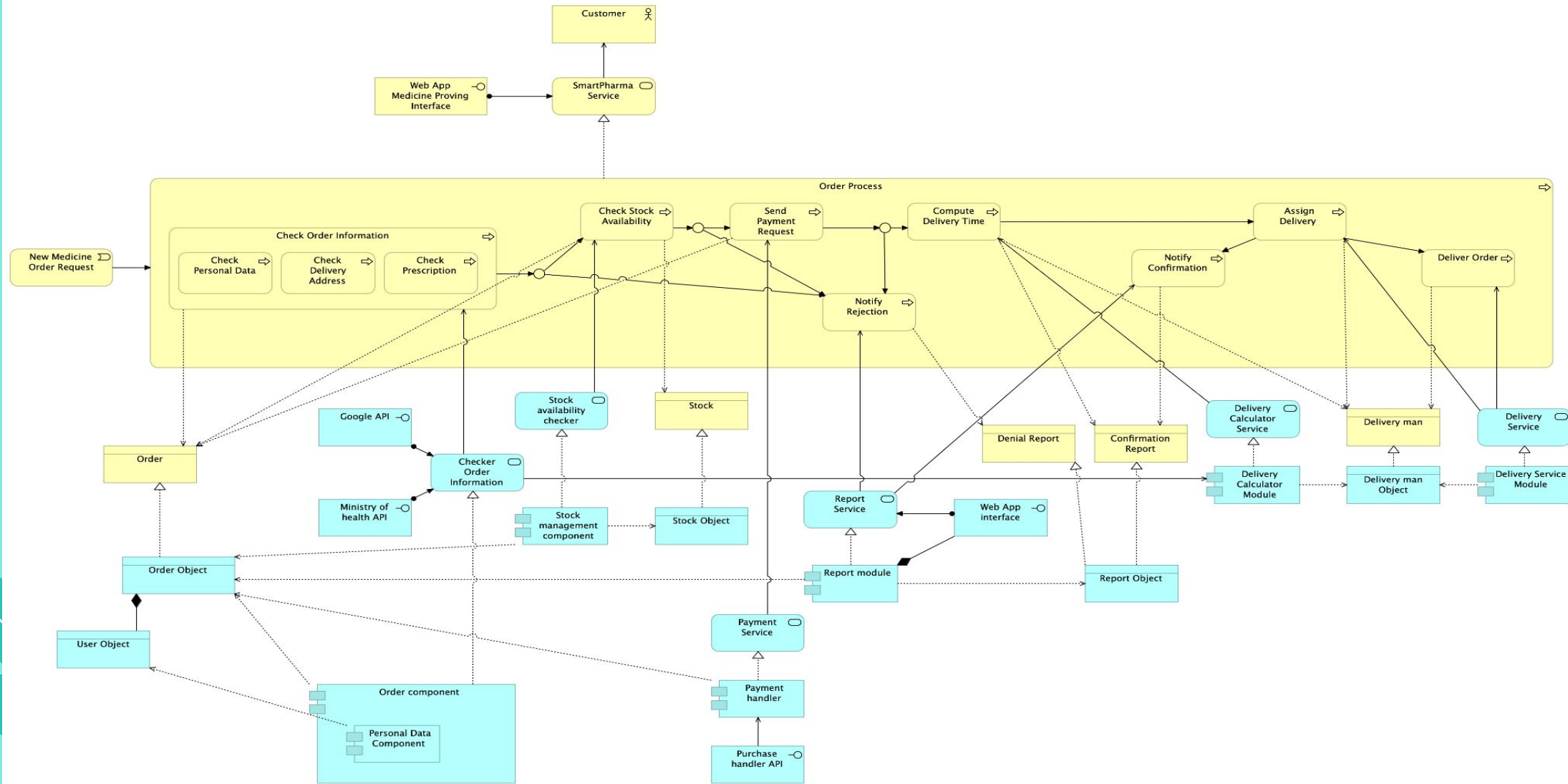
After the payment is confirmed, SmartPharma computes the delivery time by taking into consideration the maps' information and availability of delivery men.

Finally, SmartPharma contacts the selected delivery man sending her/him the order details, and provides a confirmation notification through the Web Application to the customer.



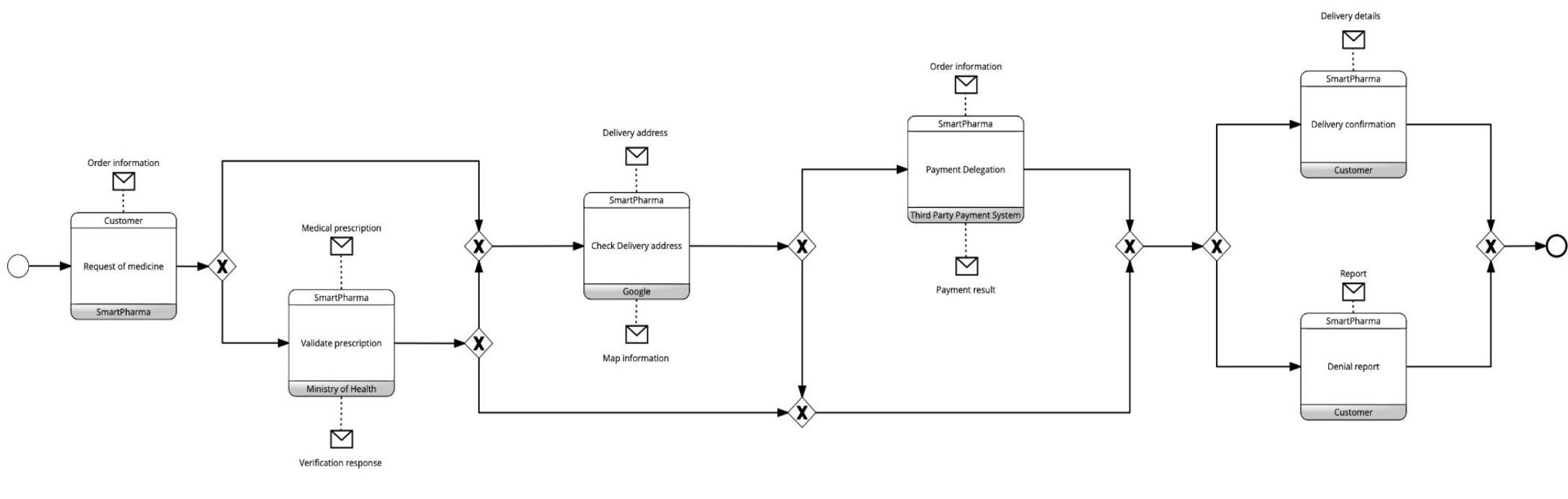
03

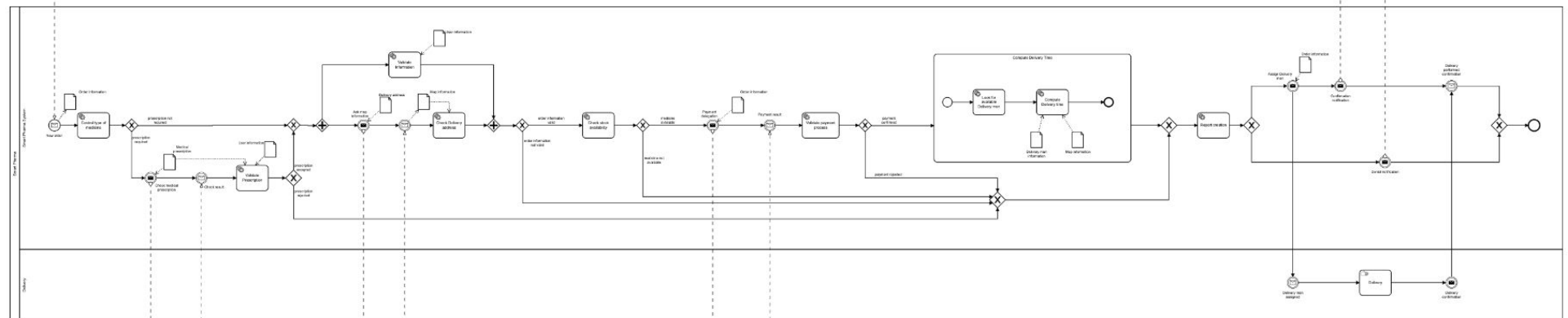
ARCHIMATE MODEL



04

BUSINESS PROCESS CHOREOGRAPHY





FROM ABSTRACT TO EXECUTABLE MODEL

01 Identify the automation boundaries

“Delivery” process, both the “Delivery” activity and the message to notify the performed activity

- The “Delivery” activity is a manual activity, therefore it can't be handled by the BPMS
- The interaction between the SmartPharma System and the Delivery stops the execution of the process until the delivery man confirms the status of the shipment

FROM ABSTRACT TO EXECUTABLE MODEL

02 Review manual tasks

The “Delivery” activity can be eliminated, since it is outside the control of the orchestrator and its activity is performed by the same actor (the delivery man) that would notify SmartPharma of the performed delivery through a User activity

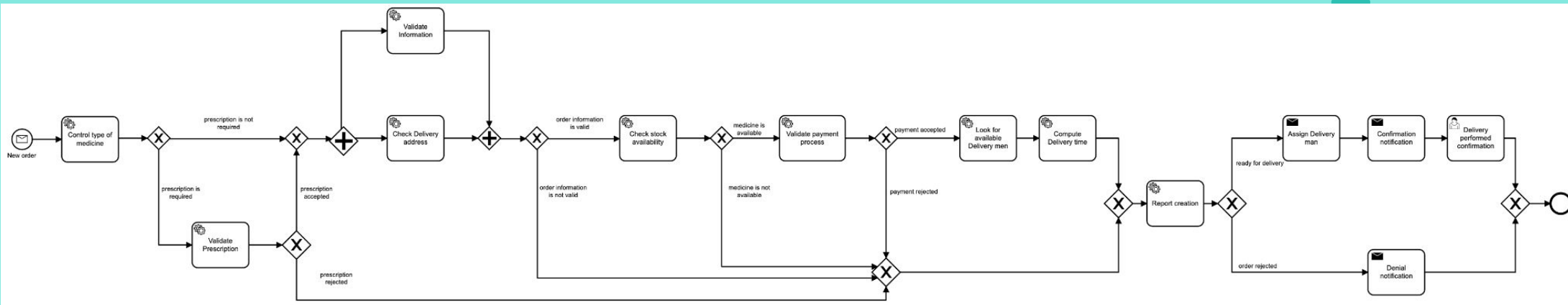


FROM ABSTRACT TO EXECUTABLE MODEL

- 03** Complete the process model
Control the execution flow
- 04** Adjust the model granularity
In the abstract model, the interactions between SmartPharma and the external actors are highlighted. In the executable model they can be incorporated in the service they enable
- 05** Specify execution properties
Set process variables, messages, forms, service details, code snippets, sequence flow

06

BUSINESS PROCESS EXECUTABLE



Medicine

Medicine*

Quantity*

Price*

Prescription

Prescription

Delivery Address

Delivery address

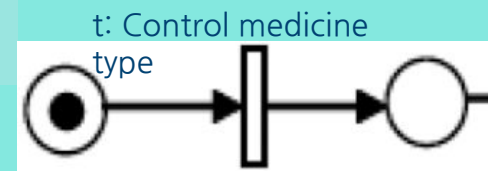
Personal Information

Name

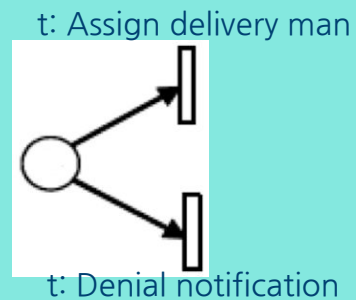
Tax Code

FROM BPMN TO PETRI NET

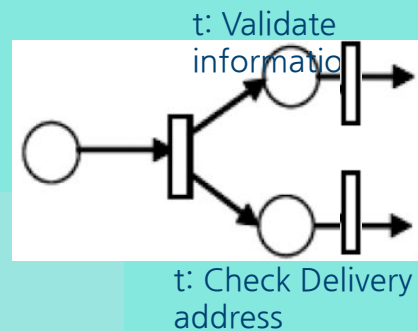
01 Activities become transitions



02 XOR



03 AND

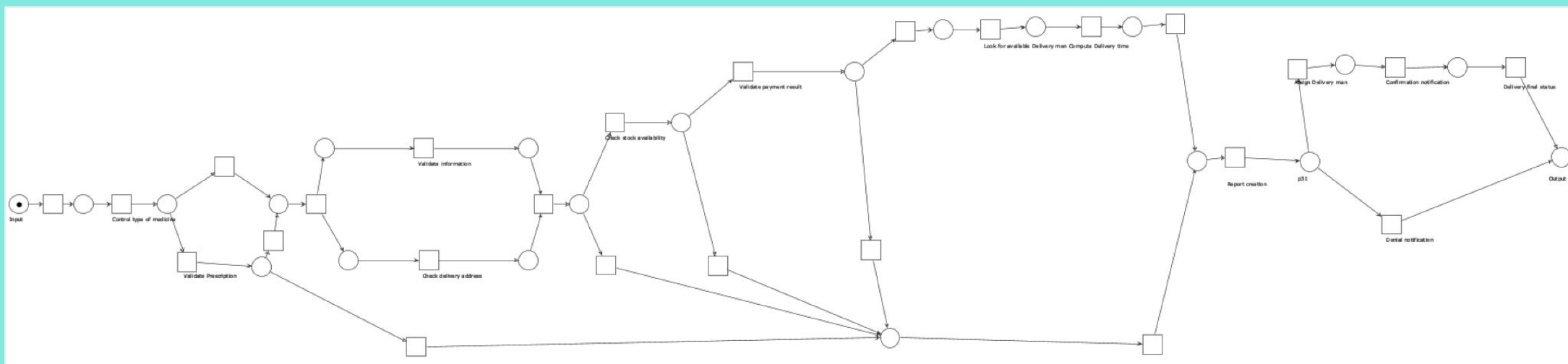


Workflow Net:

- Only one initial place
- Only one final place
- None of the nodes are not in the path from the initial place to the final place

07

PETRI NET: WORKFLOW NET

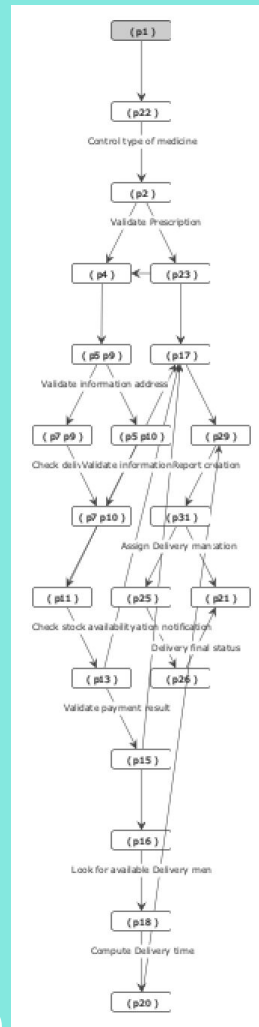


Petri Net is used to validate the semantic of a process, that means to check the absence of behavioral anomalies. For that reason, we have analyzed two properties of the Workflow Net:

- Boundness
- Soundness
- Liveness

PETRI NET: ANALYSIS

Coverability graph



Semantical analysis

Semantical analysis

Wizard Expert

Startpage

Workflow net property ☒

Soundness ☒

Net statistics

Places:	21
Transitions:	25
- Operators:	0
- Subprocesses:	0
Arcs:	52

Semantical analysis

Wizard Expert

- ☒ Qualitative analysis
 - ☒ Structural analysis
 - ☒ Net statistics
 - ☒ Wrongly used operators: 0
 - ☒ Free-choice violations: 0
 - ☒ S-Components
 - ☒ Wellstructuredness
 - ☒ Soundness
 - ☒ Workflow net property
 - ☒ Initial marking
 - ☒ Boundedness
 - ☒ Liveness

SWAGGER API DOCUMENTATION

SmartPharma API

order Everything about your order ^	
POST	/order Create a new order v 🔒 ↶
GET	/order/{order_id} Get order information by ID v 🔒 ↶
stock Operations about stock ^	
POST	/stock/inventory Check stock availability v 🔒 ↶
report Operation about report ^	
POST	/report report information v 🔒 ↶
GET	/report/{order_id} report information v 🔒 ↶
delivery Operation about delivery ^	
GET	/delivery/deliverMan Get available deliver man information v 🔒 ↶
POST	/delivery/deliverMan/{deliver_man_id} Update deliver man status v 🔒 ↶
POST	/delivery/time Calculate delivery time v 🔒 ↶
POST	/delivery/confirm/{order_id} final delivery complete confirm v 🔒 ↶

External API

https://maps.googleapis.com/ Operations about get google maps info ^	
GET	/maps/api get map information v 🔒 ↶
https://www.salute.gov.it/ Operations about validate prescription ^	
POST	/validatePrescription validate prescription information v 🔒 ↶
https://api-m.sandbox.paypal.com/ Operations about payment by paypal ^	
POST	/v2/payments/authorizations/{authorization_id} payment operation v 🔒 ↶

SWAGGER API DOCUMENTATION

API in yaml format

```
1  swagger: '2.0'
2  info:
3    description: Smart Pharma
4    version: 1.0.0
5    title: Smart Pharma
6    termsOfService: http://swagger.io/terms/
7    contact:
8      name: Aida Gasanova; Matteo Makovec; Xiaoyu Luo
9    license:
10     name: Apache 2.0
11     url: http://www.apache.org/licenses/LICENSE-2.0.html
12  tags:
13    - name: order
15    - name: stock
17    - name: report
19    - name: delivery
21    - name: https://maps.googleapis.com/
23    - name: https://www.salute.gov.it/
25    - name: https://api-m.sandbox.paypal.com/
27  paths:
28    /order:
54    /order/{order_id}:
73    /stock/inventory:
94    /report:
112   /report/{order_id}:
131   /delivery/deliverMan:
146   /delivery/deliverMan/{deliver_man_id}:
167   /delivery/time:
190   /delivery/confirm/{order_id}:
209   /maps/api:
237   /validatePrescription:
265   /v2/payments/authorizations/{authorization_id}:
```

```
292  security:
293    - smartPharm_auth: []
294  securityDefinitions:
295    smartPharm_auth:
296      type: basic
297    api_key:
298      type: apiKey
299      name: api_key
300      in: header
301  definitions:
302    Order:
343    Prescription:
350    Medicine:
360    Map:
375    ApiKey:
380    DeliveryMan:
392    DeliveryTime:
398    Report:
410    PayPalRequest:
418    PayPalResponse:
428  schemes:
429    - https
430    - http
431  basePath: /PSD/SmartPharma/1.0.0
```

SWAGGER API DOCUMENTATION

https://maps.googleapis.com/ Operations about get google maps info

GET /maps/api/{authorization_id}: get map information

Parameters

Try it out

Name	Description
authorization_id * required	get map information
string (path)	<input type="text" value="authorization_id"/>

Responses

Response content type: application/json

Code	Description
200	successful operation
	Example Value Model
	<pre>{ "lat": 0, "lng": 0, "zoom": 0, "address": "string" }</pre>
400	Invalid Prescription

Authorization control

Available authorizations

api_key (apiKey) ↗

Name: api_key
In: header
Value:

Authorize

Close

Available authorizations

Basic authorization ↗

Username:

Password:

Authorize

Close

POST /order Create a new order

Parameters

Try it out

Name	Description
body * required	Create new Order object into system
object (body)	Example Value Model
	<pre>{ "id": 0, "name": "string", "taxCode": "string", "medicineName": "string", "prescription": "string", "quantity": 0, "price": 0, "status": "placed" }</pre>
	Parameter content type
	application/json

Responses

Response content type: application/json

Code	Description
201	Created Success
400	invalid input
409	resource already exists

Demo: preparation

Start Docker

NAME	STARTED	STATUS
camunda/camunda-bpm-platform:run-latest 5eb84cf43f55 (camunda)	less than a mi	running

Config and start node js

```
const config = {  
  baseUrl: "http://localhost:8080/engine-rest",  
  use: logger,  
  asyncResponseTimeout: 10000,  
};
```

```
PS C:\Users\11945\Documents\GitHub\Published\SmartPharma\5-APIs> node ./worker.js  
✓ subscribed to topic control-medicine  
✓ subscribed to topic prescription-validation  
✓ subscribed to topic validate-information  
✓ subscribed to topic check-address  
✓ subscribed to topic check-stock  
✓ subscribed to topic validate-payment  
✓ subscribed to topic look-for-delivery  
✓ subscribed to topic compute-time  
✓ subscribed to topic report-creation  
✓ subscribed to topic assign-delivery  
✓ subscribed to topic confirmation-notification  
✓ subscribed to topic denial-notification
```

Deployed and start new instance

Deploy diagram

Deployment name

Tenant ID

REST endpoint

Include additional files +
☐ order.form

✓ Process definition deployed ×
[Open in Camunda Cockpit](#)

✓ Process instance started ×
Process instance ID: fb3ba831-f31b-11ec-b346-0242ac110002
[Open in Camunda Cockpit](#)



Demo: execution

Start Process

+

Start process

Medicine

Medicine*

Penicillin

Quantity*

2

Price*

30.5

Prescription

Prescription

Delivery Address

Delivery address

Via Cervignano 8

Personal Information

Name

Xiaoyu Luo

Tax Code

LUOXYIW2387AN250

Back

Close

Start

Log of executing

```
PRESCRIPTION IS NOT REQUIRED
✓ completed task 72cbc1df-f397-11ec-b346-0242ac110002
ADDRESS IS VALID
PERSONAL INFORMATION IS VALID
✓ completed task 72d278af-f397-11ec-b346-0242ac110002
✓ completed task 72d2519b-f397-11ec-b346-0242ac110002
Penicillin IS AVAILABLE IN STOCK
✓ completed task 73054891-f397-11ec-b346-0242ac110002
PAYMENT IS VALID
✓ completed task 73346eea-f397-11ec-b346-0242ac110002
DELIVERY MAN FOUND
✓ completed task 7362aae2-f397-11ec-b346-0242ac110002
DELIVERY TIME IS 30 minutes TO ADDRESS Via Cervignano 8
✓ completed task 73924667-f397-11ec-b346-0242ac110002
REPORT CREATED
✓ completed task 73c1badd-f397-11ec-b346-0242ac110002
DELIVERY MAN ASSIGNED
✓ completed task 73f35235-f397-11ec-b346-0242ac110002
THE ORDER CONFIRMED
✓ completed task 7422788a-f397-11ec-b346-0242ac110002
```

Demo: execution

Confirmation user task

Delivery performed confirmation

SmartPharma

Set follow-up date Set due date delivery Demo Demo

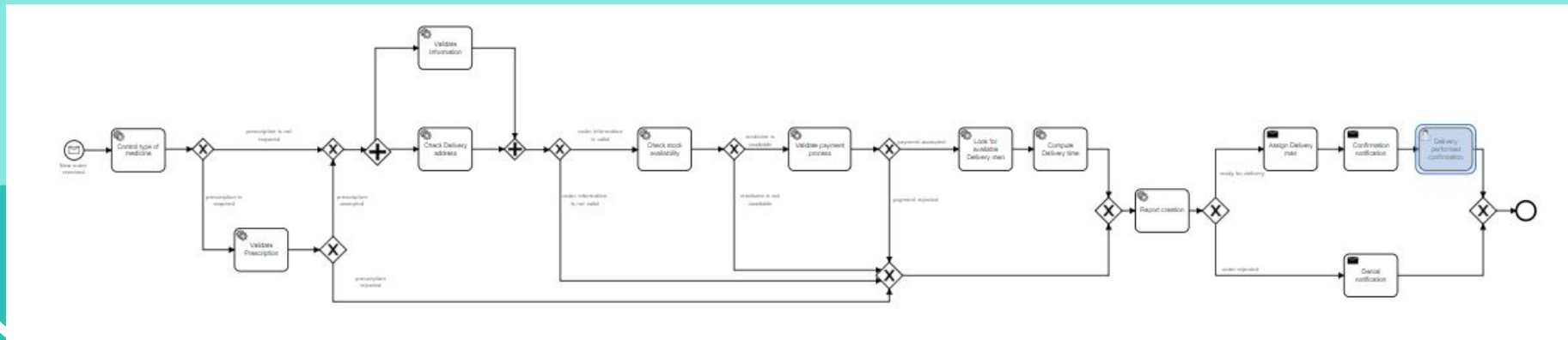
Form History Diagram Description

You can set variables, using a generic form, by clicking the "Add a variable" link below.

Add a variable +	Name	Type	Value
Remove x	status	String	complete

Load Variables

Complete





Process and Service Design Project

SmartPharma

Aida Gasanova - 10732229
Matteo Makovec - 10782774
Xiaoyu Luo - 10777140

Thanks for your attention!

