



# pharaon

PILOTS FOR HEALTHY AND ACTIVE AGEING

Grant Agreement: 857188

## D4.7

## Service Composition Support Tools – First



This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 857188.



## Document Information

|                                 |                                                         |
|---------------------------------|---------------------------------------------------------|
| <b>Deliverable number:</b>      | D4.7                                                    |
| <b>Deliverable title:</b>       | Service composition support tools – First               |
| <b>Deliverable version:</b>     | V1.5                                                    |
| <b>Work Package number:</b>     | 4                                                       |
| <b>Work Package title:</b>      | Technology support tools                                |
| <b>Due Date of delivery:</b>    | M15 (February 2021), to be amended to M21 (August 2021) |
| <b>Actual date of delivery:</b> | M23 (October 2021)                                      |
| <b>Dissemination level:</b>     | Public (PU)                                             |
| <b>Type</b>                     | Other (O)                                               |
| <b>Editor(s):</b>               | Cesar Marin (ICE)<br>Jose Luis Alfonso (ICE)            |
| <b>Contributor(s):</b>          |                                                         |
| <b>Reviewer(s):</b>             | Andrej Grguric (ENT)<br>Erika Rovini (UNIFI)            |
| <b>Project name:</b>            | Pilots for Healthy and Active Ageing                    |
| <b>Project Acronym</b>          | PHArA-ON                                                |
| <b>Project starting date:</b>   | 01/12/2019                                              |
| <b>Project duration:</b>        | 48 months                                               |
| <b>Rights:</b>                  | PHArA-ON Consortium                                     |

## Document history

| Version | Date       | Beneficiary             | Description                                                                                                     |
|---------|------------|-------------------------|-----------------------------------------------------------------------------------------------------------------|
| 0.1     | 14.10.2020 | Cesar Marin (ICE)       | Create template and ToC                                                                                         |
| 0.2     | 02.11.2020 | Jose Luis Alfonso (ICE) | Added section about ICE Orchestration                                                                           |
| 0.3     | 21.06.2021 | Cesar Marin (ICE)       | Restructured document and filled in all sections                                                                |
| 0.4     | 28.06.2021 | Cesar Marin (ICE)       | Added mock-up images and filled gaps in the text                                                                |
| 1.0     | 30.06.2021 | Cesar Marin (ICE)       | Tidied up details (captions, etc) throughout the document. Sent for internal review                             |
| 1.1     | 09.07.2021 | Cesar Marin (ICE)       | Made adjustments in response to internal reviewer's comments. Sent final version                                |
| 1.2     | 14.07.2021 | Cesar Marin (ICE)       | Added data to PM Effort table as requested. New final version produced and sent                                 |
| 1.3     | 27.09.2021 | Cesar Marin (ICE)       | Minor update in page 2. Removed data in PM Effort table as requested. New final version produced and sent again |
| 1.4     | 21.10.2021 | Cesar Marin (ICE)       | Added a "missing" section as requested by coord. New final version produced and sent again                      |
| 1.5     | 05.05.2023 | Jose Luis Alfonso       | Made adjustments in response to EC reviewer comments. Added section 1.3 as the update of resubmitted version    |

## PM Efforts per Beneficiary having contributed to the Deliverable

| #            | Partner                   | PM effort in D4.7 |
|--------------|---------------------------|-------------------|
| 1            | UNIFI                     | 0.10              |
| 28           | ENT                       | 0.10              |
| 32           | ICE                       | 3.24              |
| <b>TOTAL</b> | <b>Pharaon Consortium</b> | <b>3.44</b>       |

**Acknowledgement:** This project has received funding from the European Union’s Horizon 2020 Research and Innovation Programme under Grant Agreement No 857188.

**Disclaimer:** The content of this publication is the sole responsibility of the authors, and in no way represents the view of the European Commission or its services.

## Executive Summary

To provide to Ambient Assigned Living (AAL) related organisations with the functionalities needed in order to collaboratively define, design and orchestrate full processes and services, an appropriate software solution is needed.

In order to not re-invent the wheel, **ICE Orchestration** was selected as the software solution to provide an effective solution to support the needs for Pharaon in terms of service composition and process execution. The result will then be passed to Task 4.2 Service orchestration and customisation support tools for the actual execution of composed services. A new service composition user interface will be developed targeting non-technical users such as older adults and carers.

This is a software deliverable. Therefore, the present report is intended to be brief. The approach followed by Pharaon for service composition consists of using the ICE Orchestration tool to design processes that represent the actual composition of services.

ICE Orchestration is briefly presented focusing only on the Process Designer, Control Panel and My Tasks. A link to access ICE Orchestration is provided in Section 4. A mock-up is also presented depicting the new service composition user interface.

Finally, the next steps in Task 4.3 Service composition and support tools emphasise the new user interface and the expected feedback to be requested from the project pilots.

## Progress Beyond the State of the Art and Play

ICE Orchestration embodies the software of this deliverable. This software was created mainly for the manufacturing sector whose users are expected to be technically versed in process definition using the BPMN2.0 language. In its present form, ICE Orchestration is not suitable for non-technical users such as those found in Pharaon pilots. Therefore, and as part of the project, a new service composition user interface will be created, targeting non-technical users such as older adults and carers. A mock-up of such a user interface is presented in this document. The objective with it is to enable non-technical users to compose services without realising of the technical aspects of it. That is, the user interface will hide all technicalities and process-related terminology and the BPMN visual language. The expected result is a simplified user interface accessible by non-technical users. As far as WP4 partners are concerned, there is no such a tool with such a user interface out in the market. Not in commercial form nor as free software. **This is how Pharaon is going beyond the state of the art and play.**

## Contents

|                                                               |    |
|---------------------------------------------------------------|----|
| Executive Summary .....                                       | 5  |
| Progress Beyond the State of the Art and Play .....           | 6  |
| 1 Introduction.....                                           | 8  |
| 1.1 Overview.....                                             | 8  |
| 1.2 Update of resubmitted version .....                       | 8  |
| 1.3 How ICE Orchestration fits in Pharaon.....                | 8  |
| 1.4 Relation to other tasks and deliverables.....             | 9  |
| 1.5 Structure of the Deliverable .....                        | 10 |
| 2 Service Composition Approach for Pharaon .....              | 10 |
| 3 Technology Baseline: ICE Orchestration .....                | 11 |
| 3.1 Process designer .....                                    | 12 |
| 3.2 Code Generation .....                                     | 13 |
| 3.3 Control panel.....                                        | 13 |
| 3.4 My Tasks.....                                             | 15 |
| 4 Access to ICE Orchestration .....                           | 16 |
| 5 Mock-up of the user interface for non-technical users ..... | 16 |
| 6 Concluding remarks and next steps .....                     | 17 |

## Figures

|                                                                                                 |    |
|-------------------------------------------------------------------------------------------------|----|
| Figure 1. Simple service and user input flow .....                                              | 8  |
| Figure 2. Where ICE Orchestration fits in Pharaon architecture.....                             | 9  |
| Figure 3. Service composition approach. ....                                                    | 11 |
| Figure 4. The landing page of ICE Orchestration.....                                            | 11 |
| Figure 5. Process designer landing page. ....                                                   | 12 |
| Figure 6. Example of process composing services and human activities.....                       | 13 |
| Figure 7. Control panel showing published processes. ....                                       | 14 |
| Figure 8. Control panel showing a list of active process instances of a published process. .... | 14 |
| Figure 9. Live execution of a process instance including services.....                          | 15 |
| Figure 10. List of pending tasks for a user to complete. ....                                   | 15 |
| Figure 11. Execution of a user task.....                                                        | 16 |
| Figure 12. Mock-up of new UI for non-technical users.....                                       | 17 |

## Tables

|                                                        |   |
|--------------------------------------------------------|---|
| Table 1. Inputs from other deliverables and tasks..... | 9 |
|--------------------------------------------------------|---|

|                                                        |    |
|--------------------------------------------------------|----|
| Table 2. Outputs to other deliverables and tasks. .... | 10 |
|--------------------------------------------------------|----|

## 1 Introduction

### 1.1 Overview

This Deliverable is of type Other, which means **it is a software deliverable**. Therefore, the present document is meant to be compact, briefly explaining the approach and the technology used so far in task T4.3 Service composition support tools; it also provides a glimpse at how the technology works and details to access the live version of the primary tool used in this task. Finally, brief concluding remarks are presented also indicating future work in this task.

### 1.2 Update of resubmitted version

As part of the first version committed to European Commission and in response to the reviewer's comments, we have added the section 1.3 to better explain the role of ICE Orchestration in Pharaon.

### 1.3 How ICE Orchestration fits in Pharaon

Pharaon consists on multiple platforms and systems, each of them exposing a set of services externally (for interoperability) and internally, working together (through intraoperability). Since the aim of the project and specifically task T4.3 Service composition is to provide a tool that allows composing services and processes, ICE Orchestration has been chosen as the tool to fulfil this need.

When speaking about services, we can think on different software systems each of them requiring inputs and returning outputs. We can combine for example the output from one service and send it as the input for another service. More than often, user input is needed as well. This flow is called a process flow, or more specifically, a business process flow.

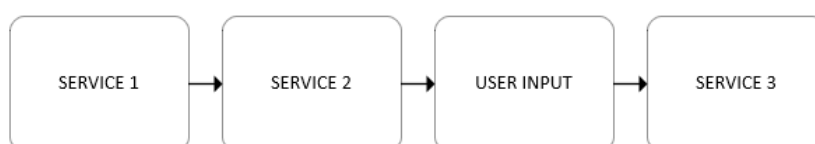


Figure 1. Simple service and user input flow

These kind of process flows are designed using BPM notation, that is composed of different visual artifacts connected, describing the flow of the data as it goes from one place (service) to other (service).

To not reinvent the wheel, ICE Orchestration has been chosen since it is a mature tool that has progressed and has been used successfully in other projects. It is open-source, free to use and it uses Camunda as its BPM engine, a mature and open-source technology.

ICE Orchestration is a complete BPM suite having at its core the open source BPMN engine from Camunda. The engine supports executing predefined flows of services (and optionally user tasks), fulfilling the Pharaon needs for service composition and orchestration.

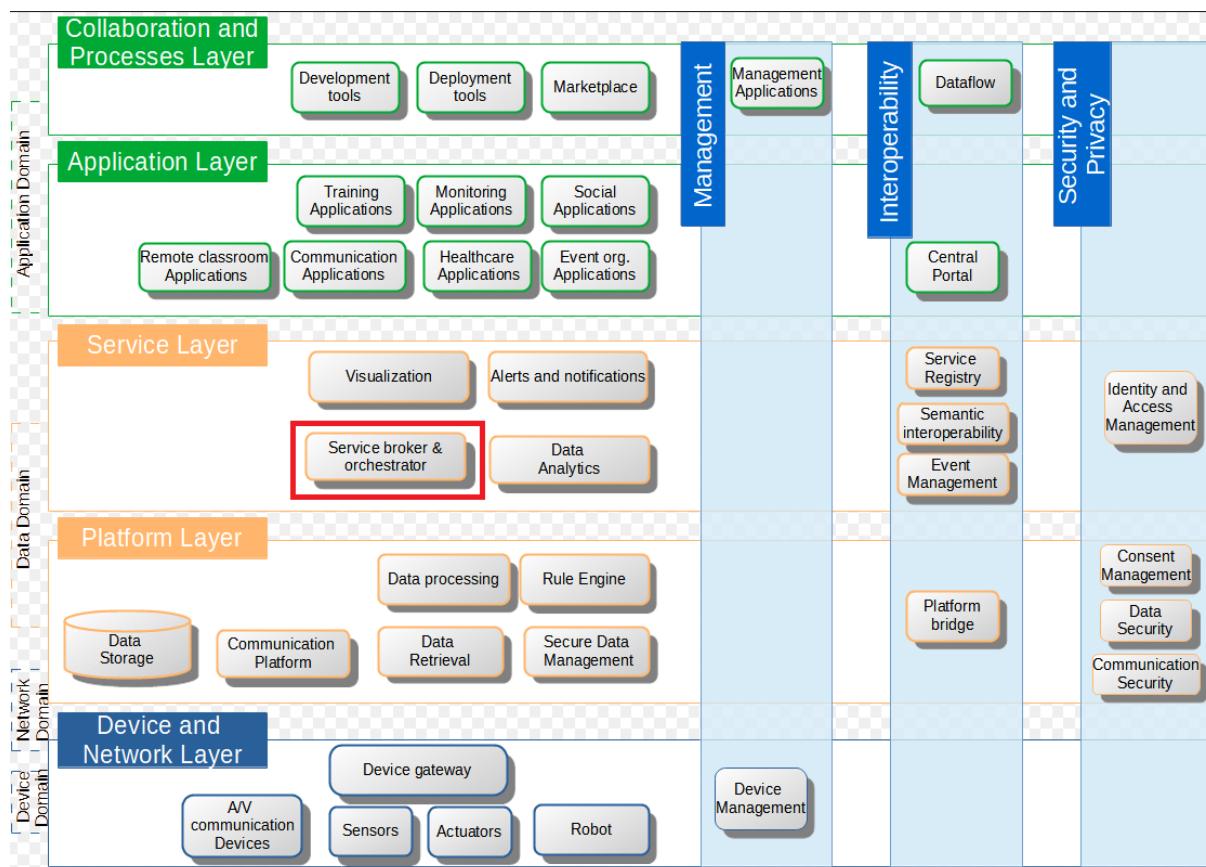


Figure 2. Where ICE Orchestration fits in Pharaon architecture

## 1.4 Relation to other tasks and deliverables

This Deliverable is related to the following other Pharaon tasks and deliverables:

**Receives inputs from:**

**Table 1.** Inputs from other deliverables and tasks.

| Input description               | Deliverable or task                         | Work package                                                     |
|---------------------------------|---------------------------------------------|------------------------------------------------------------------|
| User requirements and use cases | D2.1 User and pilot requirements            | WP2 Pilots and user requirements, initial ecosystem architecture |
| Reference architecture          | D2.2 Pharaon initial ecosystem architecture | WP2 Pilots and user requirements, initial ecosystem architecture |

**Provides outputs to:**

**Table 2.** Outputs to other deliverables and tasks.

| Output description                                                   | Deliverable or task                                   | Work package                         |
|----------------------------------------------------------------------|-------------------------------------------------------|--------------------------------------|
| Approach to service composition; ICE Orchestration tool              | D4.8 Service composition support tools – intermediate | WP4 Technology support tools         |
| ICE Orchestration and service composition UI for non-technical users | D5.1 Pharaon ecosystem – initial release              | WP5 Technology ecosystem integration |

## 1.5 Structure of the Deliverable

This report is structured as follows:

- Section 2 explains the approach followed for service composition.
- Section 3 describes the technology baseline used to realise the approach of service composition, in particular the ICE Orchestration tool.
- Section 4 indicates details of how to access the live version of ICE Orchestration.
- Section 5 provides an overview of the mock-up generated for non-technical users.
- Section 6 presents final concluding remarks and future work related to service composition.

## 2 Service Composition Approach for Pharaon

The approach to follow for service composition consists of using the ICE Orchestration tool (see Section 3 for details) for composing services. However, since this tool is too technical for non-technical users, such as older adults and carers, a new service composition user interface (UI) will be developed in Task 4.3 (see Section 5 for details). Consequently, the Pharaon service composition approach will provide two UIs to different types of users:

1. Non-technical users will use the new service composition UI; and
2. Technical users, such as Pharaon developers, will use ICE Orchestration.

Regardless of the UI used for service composition, the result will be passed to Task 4.2 Service orchestration and customisation support tools, for the actual service orchestration and execution. Figure 3 depicts the current approach in terms of software architecture. **This picture is subject to change and refinement as the project progresses.**

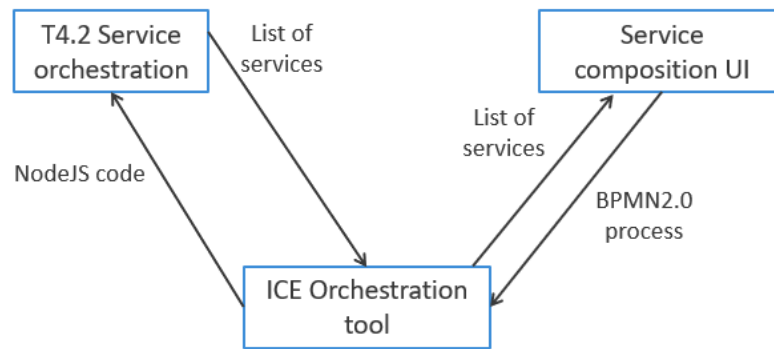


Figure 3. Service composition approach.

### 3 Technology Baseline: ICE Orchestration

ICE Orchestration is a complete workflow and service automation tool that allows users to describe and execute business processes and compose services in a visual way. A business process usually involves some work to do by humans (user tasks), work that must be done by systems (service tasks) and decision gates (gateways). There are more optional elements like throw/catch messages, script tasks, etc, that complement the process from the logic point of view. When all of these elements are put together logically and meaningfully for a business, it is called a business process or simply a process.

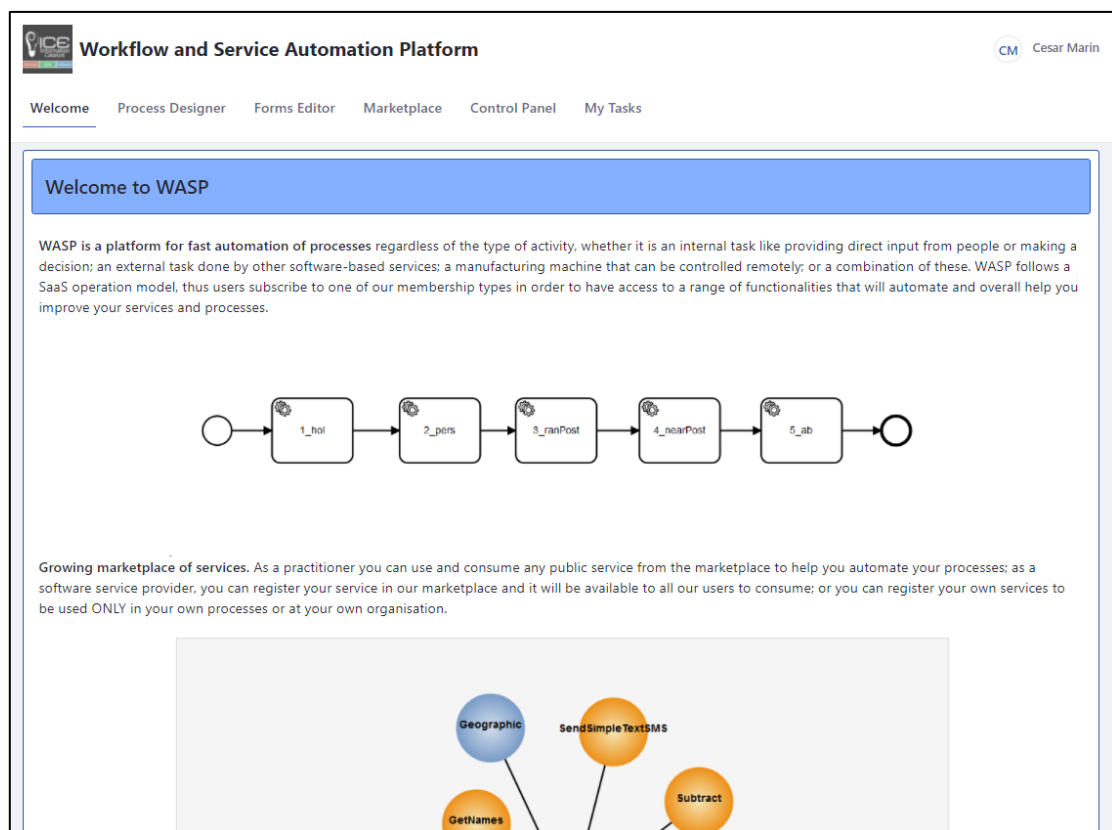


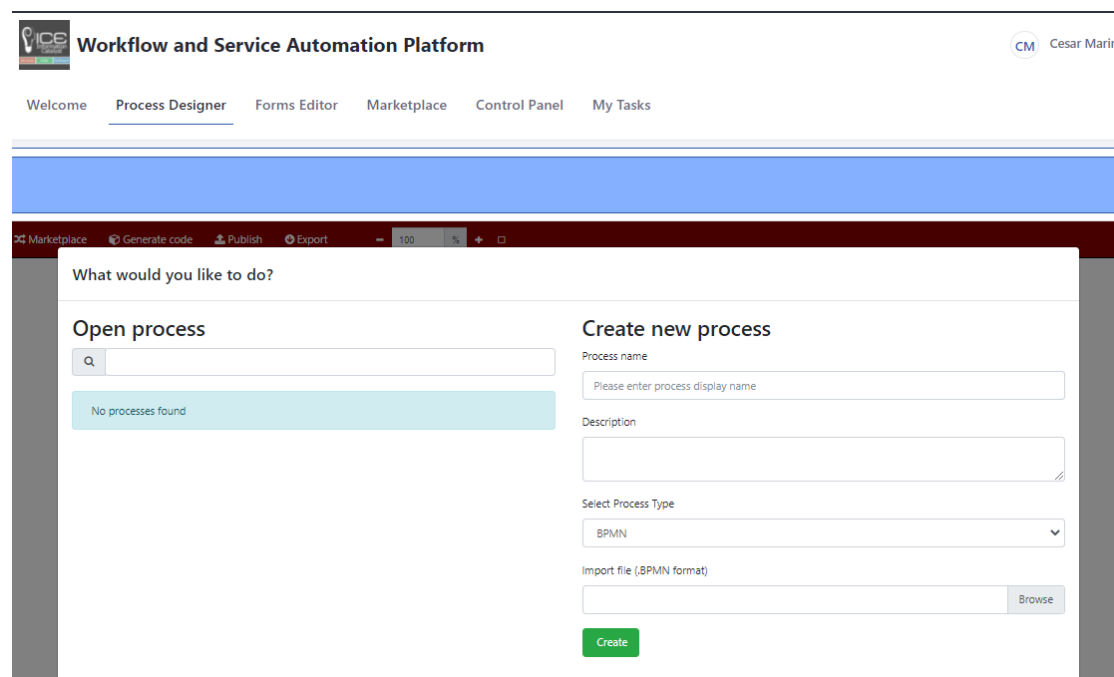
Figure 4. The landing page of ICE Orchestration.

Upon accessing ICE Orchestration, a navigation menu can be seen with different available functionalities at the top of the page. For this document, only the Process Designer, Control Panel, and

My Tasks will be explained further down. There are two major stages for any process: design and execution. In the design stage, using the Process Designer, the user graphically designs and updates process definitions. Once they are ready, the user publishes them, which means they are for execution. In the execution stage, using the Control Panel and My Tasks, the user controls the execution of defined processes. Only published processes can be executed.

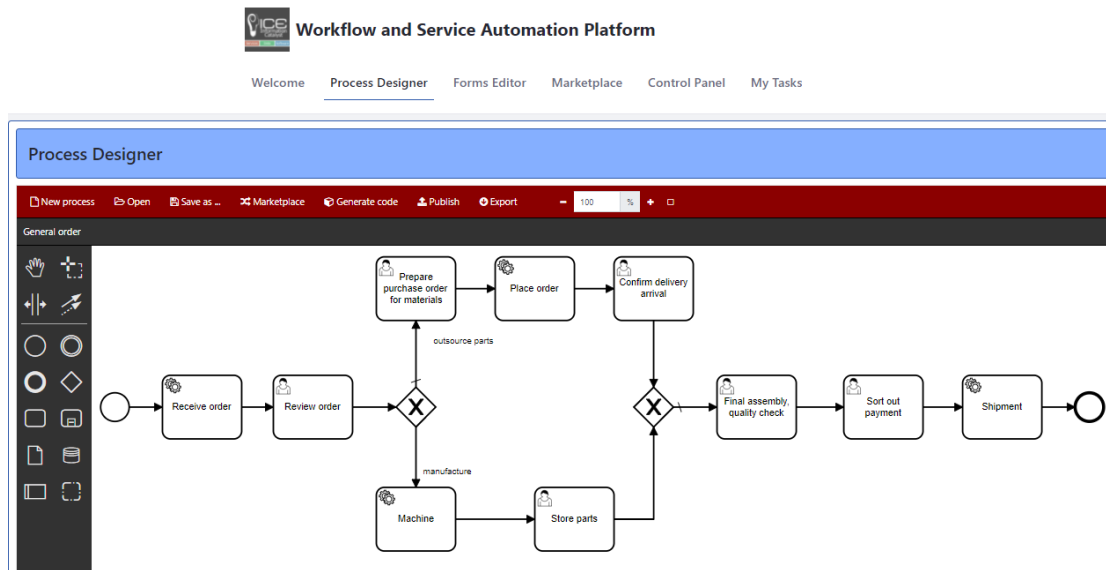
### 3.1 Process designer

The Process Designer is a key component in WASP and allows the user to design processes. It starts by asking the user to either open an existing process, import a process in BPMN format, or create a new process.



**Figure 5.** Process designer landing page.

The Process Designer comes with a palette of process elements to represent tasks, decision-making, timers, subprocesses, among many others. Each element opens up a properties panel to edit any detail related to the element itself, such as task allocation to users, definition of input and output variables per task, conditions and default execution paths. The user then can (re)design the process using the BPMN2.0 standard notation.



**Figure 6.** Example of process composing services and human activities.

The process is automatically saved with every change. To execute the process, the user clicks on the Publish button. At this point the process is validated, and any errors are highlighted to the user for immediate attention. Warnings are also indicated in a list at the bottom of the page. A process can be published with warnings, but not with errors.

### 3.2 Code Generation

ICE Orchestration supports the generation of NodeJS code based on a process diagram. The generated code can be copy-pasted inside a NodeJS Express-enabled web application for execution. The code is designed to work for processes that need fast service-to-service communication with no human intervention. Currently, this functionality is limited and needs further development, but it provides a good starting point for service orchestration outside ICE Orchestration if the need arises.

### 3.3 Control panel

The Control Panel is the main component to manage process executions. Upon landing on the Control Panel, the user sees their published processes and the number of process instances in execution, completed, and deleted. It also shows the current process status such as active or suspended. From here, the user can launch the execution of new process instances and inspect the details of process instance executions by clicking on Manage Process Instances.

**Workflow and Service Automation Platform**

CM Cesar Marin

Welcome Process Designer Forms Editor Marketplace **Control Panel** My Tasks

**Control Panel - Process List**

| Process Name  | Process Owner | Deployment Time     | Instances Running | Instances Completed | Instances Deleted | Process Status |
|---------------|---------------|---------------------|-------------------|---------------------|-------------------|----------------|
| General order | Cesar Marin   | 10:19:06 03/06/2021 | 1                 | 0                   |                   |                |
| supply2       | Cesar Marin   | 07:52:21 11/09/2020 | 2                 | 2                   |                   |                |

Actions: + Start Process Instance, Manage Process Instances, Suspend Process, Redeploy Process

Refresh

**Figure 7.** Control panel showing published processes.

A new list appears showing the active process instances of a particular process definition. From here, the user can visualise the live execution.

**Workflow and Service Automation Platform**

CM Cesar Marin

Welcome Process Designer Forms Editor Marketplace **Control Panel** My Tasks

**Control Panel - Process Instance List**

Process Name: General order Active

Return to Process List

| Process Name  | Instance Owner | Start Time          | Finish Time | Instance Status |
|---------------|----------------|---------------------|-------------|-----------------|
| General order | Cesar Marin    | 10:20:27 03/06/2021 |             |                 |

Actions: + Start Process Instance, View Process Instance, View Process Instance Activity, View Process Instance Variables, Suspend Process Instance, Delete Process Instance

Refresh

**Figure 8.** Control panel showing a list of active process instances of a published process.

Completed tasks appear in a faded colour. Active tasks appear in bright green whereas cancelled tasks appear in red. Tasks in white have not yet been enabled. Process variables are also inspectable by hovering the mouse over any task.

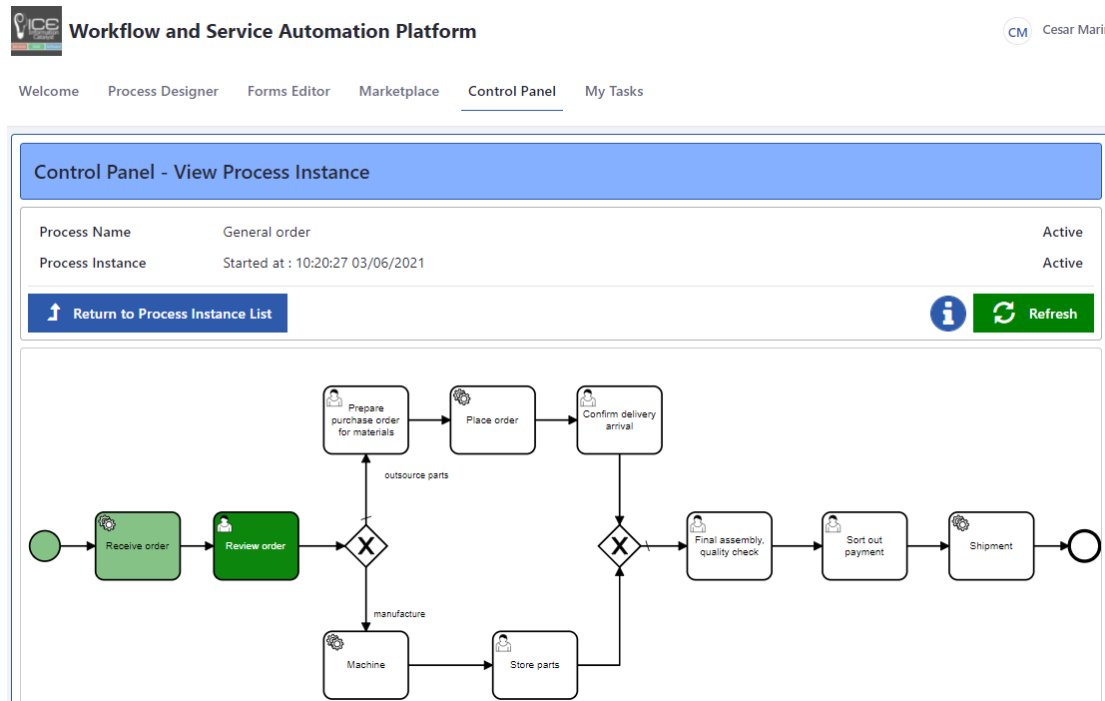


Figure 9. Live execution of a process instance including services.

### 3.4 My Tasks

The final key component for process execution is called My Tasks. On this page, users can see a list of their active tasks from all processes they have been allocated. Only User Tasks appear in this list. The user decides which task they want to execute next by clicking on the Start Task button.

**Workflow and Service Automation Platform**

Welcome Process Designer Forms Editor Marketplace Control Panel **My Tasks**

**My Tasks - Task List** [Refresh](#)

| Process Name  | Process Owner | Task Name                            | Task Assignee | Start Time          | Task Status |                            |
|---------------|---------------|--------------------------------------|---------------|---------------------|-------------|----------------------------|
| General order | Cesar Marin   | Review order                         | Cesar Marin   | 10:20:28 03/06/2021 | Running     | <a href="#">Start Task</a> |
| supply2       | Cesar Marin   | Confirm delivery arrival             | Cesar Marin   | 17:31:10 11/09/2020 | Running     | <a href="#">Start Task</a> |
| supply2       | Cesar Marin   | Prepare purchase order for materials | Cesar Marin   | 09:17:32 11/09/2020 | Running     | <a href="#">Start Task</a> |

Figure 10. List of pending tasks for a user to complete.

Once a task starts its execution, the user has to complete the form. This form is generated based on the input and output variables assigned to the task in the Process Designer. Once the user is happy with the inserted information, they click the Finish button and the task is completed. The process instance then moves to the next task in the execution path.

**My Tasks - Task Form**

Process Name: General order      Active  
 Process Instance: Started at : 10:20:27 03/06/2021      Active  
 Task: Started at : 10:20:28 03/06/2021      Running

[Return to Task List](#) [Refresh](#)

| Task Name    | Task Assignee | Start Time          | Task Status |
|--------------|---------------|---------------------|-------------|
| Review order | Cesar Marin   | 10:20:28 03/06/2021 | Running     |

[View Process](#)

**Task Description**  
 Example of an order for parts

Please complete the following form

**Task Inputs**

| Input Name | Input Value |
|------------|-------------|
| amount     | 53.254761   |
| address    | \$43 3SU    |

**Task Outputs**

| Output Name | Output Value         |
|-------------|----------------------|
| full_name   | <input type="text"/> |

[Finish](#)

Figure 11. Execution of a user task.

## 4 Access to ICE Orchestration

ICE Orchestration can be found live at <https://pharaon.icelab.cloud/> as the software part of this Deliverable. Due to security reasons and the public nature of deliverable D4.7, ICE cannot disclose any account (username and password) in this document. However, dedicated credentials to access the tool and explore it can be generated upon request.

The live version of ICE Orchestration is considered the production version for Pharaon and is ready for integration with the rest of the Pharaon ecosystem. It is also relevant to highlight that it will change as the project progresses. Therefore, the actual UI could look different from the screenshots shown in this document in the future.

## 5 Mock-up of the user interface for non-technical users

As described in Section 2, a new service composition UI will be created for Pharaon, targeting non-technical users such as older adults and carers. A mock-up of such UI was generated after a series of discussions with partners. Figure 12 presents snapshots of the generated mock-up. The objective with it is to enable non-technical users to compose services without realising of the technical aspects of it. That is, the UI will hide all technicalities and process-related terminology and the BPMN visual language. The result is a UI simple to use by anybody. **This is an innovative element in Pharaon.**

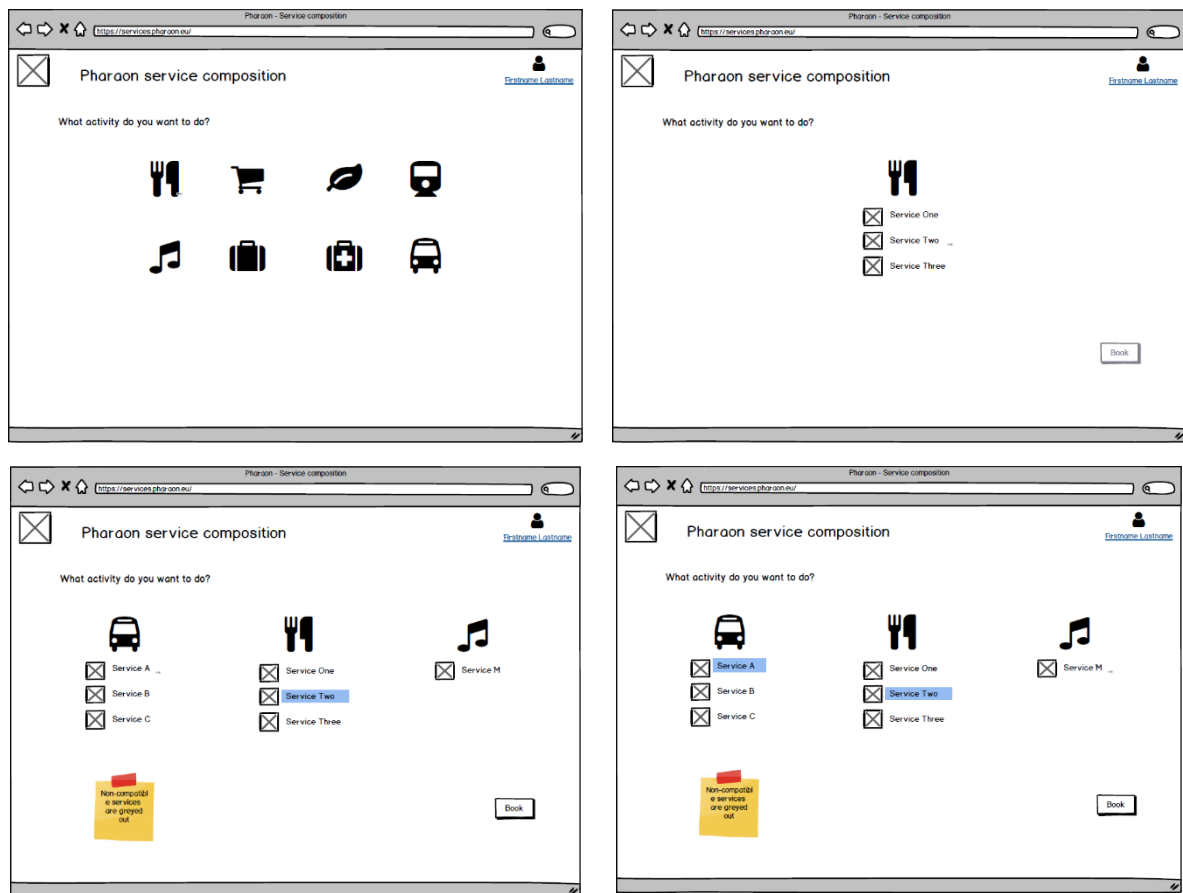


Figure 12. Mock-up of new UI for non-technical users.

## 6 Concluding remarks and next steps

This deliverable (ICE Orchestration software and this brief report) marks the initial steps towards service composition across the Pharaon ecosystem. As such, the approach for service composition in Pharaon has been presented. ICE Orchestration tool has also been introduced and a link has been provided to visit the live version of it. Finally, a mock-up for a new service composition UI targeting non-technical users has been introduced.

The immediate next steps along this line of work consist of the following, in no particular order:

- Adjust the mock-up according to the use cases in each pilot and ask them for feedback.
- Refine the approach architecture.
- Connect ICE Orchestration to other parts of the approach architecture and to the rest of the Pharaon ecosystem.
- Start development on the new service composition UI.