Reflow | Co-Creating Circular Resource Flows in Cities

constRuctive mEtabolic processes For materiaL flOWs in urban and peri-urban environments across Europe

Deliverable 2.1

Use Case Analysis and Requirements

Due date of deliverable: 31/05/2020
Actual submission date: 29/05/2020
Start date of project: 01/06/2019 Duration (36 Months)
Dissemination Level: Confidential ✓

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

DELIVERABLE

Work Package	WP2 IT Infrastructure and Tools
Deliverable	D2.1 Use Case Analysis and Requirements
Task(s)	Task 2.1 Use Case Analysis and Requirements [M1-M12]
Document Name	Use Case Analysis and Requirements
Due Date	M12: 31.05.2020
Submission Date	29.5.2020
Dissemination Level	[X] P – Public [ ] CO – Confidential
Deliverable Lead	Fraunhofer FOKUS
Author(s)	Lina Bruns (Fraunhofer FOKUS), Benjamin Dittwald (Fraunhofer FOKUS), Ivan Minutillo (Dyne), Pablo Muñoz Unceta (IAAC)
Point of Contact	Lina Bruns, lina.bruns@fokus.fraunhofer.de
Reviewers	Cecilie Schulze (DDC), Christian Villum (DDC), Malene Jung (DDC), Ista Boszhard (Waag), Taco van Dijk (Waag), Cecilia Raspanti (Waag)
Status	[ ] Plan [ ] Draft [ ] Working [x] Final [ ] Approved
Abstract (for public dissemination only)	This document contains the use cases and requirements for the technical tools that support the pilot cities in Reflow becoming more sustainable by incentivising circular practices in local ecosystems through monitoring and optimization of urban metabolic processes. This document describes the effort undertaken within the context of Task 2.1 *Use Case Analysis and Requirements* and points out in detail the methodology used to create the use cases and to collect the requirements for both tools, ReflowOS and the Open Data Dashboard.
Keywords	Requirements, User Stories, Use Cases

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Statement of Originality	This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Revision History

Revision	Date	Author(s)	Organization	Description
D2.1 v0.1	25 February 2020	Lina Bruns	Fraunhofer	First draft
D2.1 v0.2	3 March 2020	Aspasia Beneti	Dyne	Additional parts added, first review
D2.1 v0.3	9 March 2020	Lina Bruns	Fraunhofer	Corrected version after feedback, scenarios and introduction added
D2.1 v0.4	29 March 2020	Lina Bruns	Fraunhofer	First set of requirements added, User stories and personas
D2.1 v0.5	14 April 2020	Pablo Muñoz Unceta, Guillem Camprodon	IAAC	Additional parts added, review
D2.1 v0.6	16 April 2020	Lina Bruns	Fraunhofer	Corrected version after feedback, Cluj scenario added
D2.1 v0.7	20 April 2020	Ivan Minutillo	Dyne	Improved scenarios and requirements – general review
D2.1. v0.8	22 April 2020	Lina Bruns	Fraunhofer	Corrected version after feedback, glossary added
D2.1 v0.9	28 April 2020	Benjamin Dittwald	Fraunhofer	Added Open Data Dashboard specific texts
D2.1 v1.0	30 April	Lina Bruns, Benjamin Dittwald	Fraunhofer	Use Cases added, final review
D2.1 v1.1	20 May 2020	Lina Bruns, Yury Glikman	Fraunhofer	Addressing the comments of the reviewers

The information, documentation and figures in this deliverable are written by the REFLOW project consortium under EC grant agreement number 820937 and do not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Table of Contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

List of Figures

List of Tables

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

List of abbreviations and key words

Agile Methodology The Agile Methodology comes from software development and focuses on incremental delivery, continual planning and continual learning.

API Application programming interface

Business process A business process is a collection of structured activities that achieve a specific goal.

Circular Economy (CE) A Circular Economy is an alternative to a dominant linear industrial model of design, produce, purchase, consume and dispose. A circular model aims to redefine growth and a positive societal impact. For its development a systemic approach and a deep transformation of habits and behaviour are needed. It entails a transition from using finite energy resources, to using renewable ones (designing the concept of-waste out of the system), while building economic, natural and social impact. Although starting from different materials in REFLOW, the focus of the circular economy gradually extends beyond these issues related to material management and covers other aspects such as the social impact, technological aspects and the evolution of urban governance structures. Since Circular Economy is purposive, but isn't just about sustainability environmentally and economically, the social components are fundamental for this transition to happen. This happens in mindset and cultural rituals: from the way we design, produce, consume, purchase and dispose, all the way to how we see and value existing, or even abundant, resources present in the local environments.

Co-creation Co-creation is a product or service design process that integrates users.

Core requirement Core requirements are requirements describing the behaviour of the economic foundation. These are the requirements for functionalities that ReflowOS should offer and will be available for each pilot.

Data dashboard A data dashboard is an information management tool that tracks, analyses and displays key data points.

Design Thinking Design Thinking is a human-centred approach towards (complex) problems which are seen and (re)defined as design challenges. In order to tackle these challenges, a non-linear, iterative process seeks to understand users, challenges assumptions, redefines problems and creates solutions to prototype and test in its actual context. Design Thinking consists of five phases that are extracted from a creative design process and translated/ captured in five simplified steps that can be run through simultaneously: Empathise, Define, Ideate, Prototype and Test. Design Thinking is both a mind-set, a method and an approach and can be applied in a variety of fields.

Ecosystem A (business) ecosystem is a network of organizations involved in the delivery of a product or service through competition and cooperation.

Front-end Front-end is the client-side or web design in web industry

Functional requirements	Functional requirements define the required behaviour of the system to be built. They define the capabilities that a product must provide to its users. Functional requirements are based on system objectives and respond to the critical task of ensuring the right implementation of the expected functionality in the final software.
GUI	Graphical user interface
Metadata	Metadata is structured data that contains information on characteristics of other data.
Minimum viable product (MVP)	A minimum viable product is a version of a product with just enough features to attract early users and validate a product idea early in the product development cycle.
Non-functional requirement	Non-functional requirements specify other characteristics of ReflowOS, other than functionality. These requirements can be subcategorized into categories such as performance, design constraints, etc. Non-Functional requirements can also describe quality attributes or implementation constraints that the product must have. Thus, they are more qualitative than functional requirements.
Open data	Open data is the idea of data that can be used and redistributed by everyone for every purpose. Restrictions are only made to clarify the source or to secure openness of the data.
Persona	A persona is a fictional character presenting a user type that might use the product. It is created based on research and helps to understand the users' needs, experiences, behaviours and goals
Pilot requirement	Pilot requirements are pilot specific requirements that go beyond the economic foundation.
Requirements elicitation	In requirements engineering, requirements elicitation is the process of collecting information to research and discover the requirements of a system.
UI	User interface
Urban metabolism	Urban metabolism can be understood as the collection of complex socio-technical and socio-ecological processes by which flows of materials, energy, people, and information shape the city, service the needs of its populace, and impact the surrounding hinterland.
Use case	A use case gathers possible scenarios of interactions between an actor and a system to achieve a business goal.
User requirement	User requirements describe what needs a user has when using a system, e.g. which activities the user must be able to perform.
User stories	A user story is a tool often used in Agile software development, describing a feature of the software from the end user's perspective. As user stories clearly state the different types of users, what they desire and why, they can be seen as a very high-level definition of requirements. A user story is short and consists generally of one sentence only. Still, user stories contain enough information

about the requirement, so that developers can work with them and estimate the effort needed to implement the feature.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement number 820937.

1. Introduction

1.1 Objectives of the Deliverable

The scope of D2.1 is to document the preliminary efforts undertaken within the context of Task 2*.1 Use case analysis and requirements*. The aim of T2.1 is to analyse use cases and to collect and analyse user requirements for ReflowOS and the Open Data Dashboard. For this, the requirements extraction techniques of agile software development are being followed, adopting the principles of user stories. The user stories (and subsequently the elicited requirements) stem directly from the pilots, leading to the collection and analysis of both core as well as pilot specific requirements. The user stories, containing both functional and non-functional requirements, include a complete high-level description of the expected behaviour of the system that is going to be specified and developed and build the basis for the implementation of the platform's minimum viable products (MVP). The deliverable will maintain the requirements backlog in order to guide all future development tasks (T2.2 – T2.6). However, the collection of the requirements and especially their periodization will follow an iterative process, so that this deliverable can only document the status of requirements after twelve months.

D2.1 is an internal document meant for all consortium members interested in the requirements of the two tools that are going to be developed in the scope of work package 2. For reading and understanding this document no technical knowledge is needed. The methodology used to derive the requirements is explained clearly and in detail.

1.2 Structure

Deliverable 2.1 is organized in five main sections as indicated in the table of content.

- The first section briefly introduces the deliverable. It documents the scope of the deliverable and its objectives. It also documents the starting situation and thus clearly shows which effort has been undertaken in the scope of Task 2.1 *Use Case Analysis and User Requirements*. This section also describes how this document is structured.
- Following the introductory section, Section 2 describes the methodology that was followed for the identification of the requirements. The deliverable documents the various steps followed for the engineering of the requirements in the context of the project, and how the user stories collected were translated into functional and non-functional requirements.
- Section 3 is split into two parts: 3.1 introduces three general scenarios that show how pilots can use ReflowOS in future, while Section 3.2 presents a scenario for the Open Data Platform. These scenarios have been developed in order to give the pilots more guidance and a vision of what will be possible with ReflowOS and the Open Data Platform. These general scenarios have been adjusted to the respective pilots needs in form of pilot scenarios, which are also introduced in this section.
- Section 4 documents the functional and non-functional requirements that were derived out of the user stories. These requirements will actually drive the design and implementation of ReflowOS and the Open Data Platform.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

- Section 5 concludes the deliverable. It outlines the main findings of the deliverable which will guide the future research and technological efforts of the consortium.

Deliverable 2.1 includes also a series of annexes:

- 6.1 User Stories
- 6.2 Pilot specific characteristics
- 6.3 Personas
- 6.4 Use Cases

1.3 Vision, starting situation and approach

1.3.1 Vision

The vision of Reflow is to develop circular and regenerative cities through the re-localisation of production and the re-configuration of material flows by providing best practices of circular principles that can easily be adopted by cities. The best practices are being developed within six pilot cities by showing examples of ways in which new business models for circular economy (CE) can be created and adopted. In order to support the pilot cities with this task, different tools are going to be developed within the project scope. One of these tools is ReflowOS, a solution that helps incentivising the circular practices in local ecosystems by monitoring and optimisation of urban metabolic processes. Concretely, ReflowOS is planned to be a peer-to-peer and secure economic network that allows online economic activities: monitoring, track and tracing, and coordination among participants without central control. Work package 2 includes the activities for developing ReflowOS. In the scope of Task 2.1, use cases for the use of ReflowOS are analysed and requirements from the user perspective identified. These activities prepare the actual software development and are necessary in order to develop a product that meets the users' needs in the end.

1.3.2 Starting Situation

Starting the development of prototypes we rely on already existing technologies to avoid redundancy. The following table shows the background technology we use in WP 2 to build the Reflow IT tools upon.

Table 1: Background Technologies used in WP2.

Background Technology	Short Description
React	JavaScript library for building user interfaces. It is maintained by Facebook and a community of individual developers and companies. It can be used as a base in the development of single-page or mobile applications.
Styled Components	Javascript library for building reusable UI modules across multiple websites.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Graph QL	Data query and manipulation language for webservices APIs, and a runtime for fulfilling queries with existing data.
ValueFlows	Value Flows is a set of common vocabularies to describe flows of economic resources of all kinds within distributed economic ecosystems.
ZenRoom	Crypto VM for database and blockchain
ActivityPub	ActivityPub is a decentralized social networking protocol based on the ActivityStreams 2.0 data format.
Vue.js	Vue.js is a JavaScript library for building user interfaces. It is used in order to build single-page applications.
Vert.X	Vert.X is a Java tool set in order to create complete asynchronous applications.

1.3.3 Approach

For the development of the Reflow IT tools, we followed the agile methodology (see Section 2.1). At the very beginning, interviews were conducted with each pilot to learn about their ideas, visions, needs and stakeholders. The interviews took place in M5 and the results were documented for each pilot. This documentation was the starting point for the online Pilot Cities Framework set up in context of WP 5.

From the interviews, we learned that some pilots were already more advanced with developing their project scenario while others were quite at the beginning, with less concrete ideas of how an IT solution could support their local CE activities. Thus, based on the results of the interviews, we developed four general scenarios that indicate what the Reflow IT tools could be used for (see Section 3.1). These scenarios were presented to each pilot at the first co-creation workshop in Copenhagen in November 2019. Showing them what is possible to implement from a technological perspective gave the pilots more guidance and a clearer idea of what the tools could be used for. With two more advanced pilots, the general scenarios were transformed in pilot specific scenarios by adjusting them to their local ecosystem (see Section 3.2).

After the first co-creation workshop it became obvious that the pilots are interested in different scenarios of the Reflow IT infrastructure. We identified two groups: Group A are pilots interested in developing an economic decentralized network, the so-called ReflowOS, which can facilitate material exchange, real time material localization and tracing material flows. Group B are pilots that would like to have a data dashboard that enables them to publish, visualize and enrich their data as open data, so that the community can reuse it (the so-called Open Data Platform). Both scenarios are based on data and information about the corresponding resources. Therefore, we decided to build an economic foundation as a common ground for each of the pilot: the track-and-trace scenario described in Section 3.1. This enables pilots to track and trace the material flow in the ecosystem: they can register the transfers of material from one place to another as well as the change of ownership of material and thus produce data about their material.

After communicating the different possible scenarios to the pilots and refining them with two pilots, further workshops took place to identify the pilots' needs and develop first prototypes of the solution. As we identified two groups being interested in two diverse solutions, we decided to have two different focus sessions. This allowed us to concentrate especially on the desired functionalities while paying attention to the local conditions within the pilots. In each of the two workshops, the participants identified user stories, which can be seen as high-level requirements (see Section 2.3).

In the next step, the user stories were assembled and refined and first requirements were documented. During the online project meeting in Vejle in March 2020, the intermediate results were presented to the pilots. The results of these workshops were documented and used to deviate further requirements. In parallel, the user stories were used to create use cases. Use cases show possible interactions between a software and its user and thus helps us to create a more advanced vision of ReflowOS and the Open Data Dashboard. The use cases can be found in Section 6.4. The use cases were also used to derive more requirements for both ReflowOS and the Open Data Dashboard. The requirements are documented in section 4. However, one has to keep in mind that the listed requirements represent the current status and progress we have achieved so far. It is assumable that the requirements might change over the remaining course of the project run time, or that new requirements will be found.

2. Methodology of Requirements Identification

2.1 Description of the Methodology

The software development methodology that seemed to be most suitable with the REFLOW purposes and needs is the Agile development. Agile software development focuses on incremental delivery, continual planning and continual learning. In contrast to the classical waterfall software development, in agile there is no huge theoretical concept being developed in the beginning of the project that is tested for its practical suitability after months of development. Instead, in Agile, the requirements are defined in an iterative process and re-evaluated every iteration according to external changes, resources and time remaining.
This is especially suitable for Reflow, since the pilots first had to develop their project scenarios, and thus the requirements could not be fully collected at the beginning of the project.

Agile development consists of many continuous improvement cycles, so that first prototypes and MVPs are already developed at a very early stage of the project. These early results go through a number of iterations before anything is final. As feedback is gathered and implemented continually, the method is very communication-oriented and people focused.

As with the agile methodology, the development is done in many small iterations, quick reactions to changing requirements are possible. The agile methodology accepts that change cannot be avoided and requirements, research and development happen in parallel. In addition, the iterations provide opportunities to assess the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

direction of a project throughout the development lifecycle. The agile methodology is a good addition to the design thinking approach we also followed when creating personas and user stories: Design Thinking explores the problem and focuses on understanding the users’ needs while the agile methodology describes the way how we adapt to changing conditions within the software implementation.

Figure 1 shows the agile methodology used in Reflow.

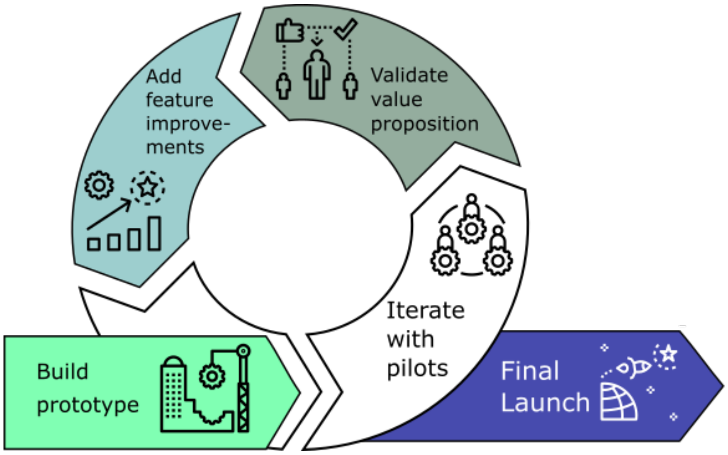


Figure 1: Agile methodology in Reflow.

The main benefits of the agile approach are:

- Early results: the minimum viable product is generally built first, demoed and tested as soon as possible. The feedback received is taken as an input to the following cycle and the incremental part of the software.
- Adaption to changes: developers can react to changes when new input comes in because the full software is designed in incremental modules.
- Predictability: Issues, delays and other unplanned changes can be foreseen easier because the progress is reviewed in short periods.
- Close collaboration: the continuous feedback cycles lead to a close collaboration between the development team and the business stakeholders. Thus, risks in terms of lacking communication are minimized.

As mentioned before Agile development emphasizes close collaboration between the business stakeholders and the development team. In accordance with that, the requirements elicitation was achieved adopting the principles of user stories (see Section 2.3).

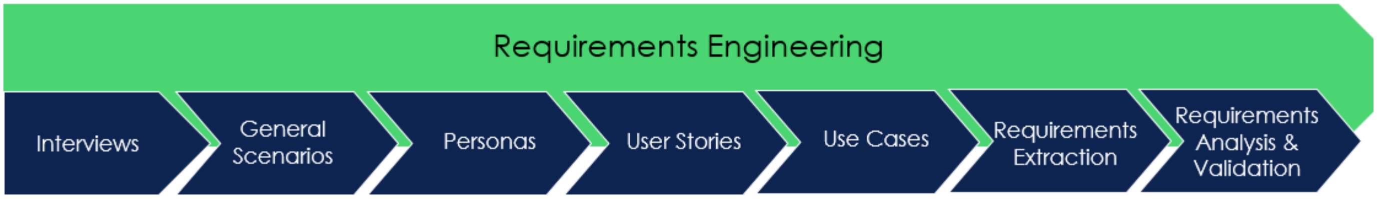


Figure 2: Requirement Engineering in ReflowOS.

Figure 2 represents the phases for the definition of user requirements for ReflowOS. At the very beginning, interviews were conducted with each pilot to identify the stakeholders involved, their needs and basic ideas for the pilot scenarios. This has taken to the definition of general scenarios for ReflowOS (see Section 3.1.1 and the Open Data Dashboard (see Section 3.1.2) and to the definition of first high-level goals, objectives and needs of the organization in step 2. Based on these first ideas of both tools, some of the pilots created personas (see Section 6.3) showing the challenges and problems of archetype users. From this, the pilots then created user stories (see Section 6.1), showing which objective a particular user wants to achieve through the interaction with the respective system. Both the creation of personas and user stories were performed in guided sessions with the technical partners and some of the pilots (Amsterdam, Cluj, Vejle, Paris).

Based on the personas and user stories, use cases (see Section 6.4) were created by the technical partners that precise the interaction between the user and the respective system in step 5. In the next step we extracted functional (see Section 4.1) and non-functional requirements (see Section 4.2) from the results of the user stories and use cases. Finally, the technical partners analysed the extracted requirements for consistency.

2.2 Persona Creation

A persona is a fictional character presenting a user type that might use the product. It is created based on research and helps to understand the users' needs, experiences, behaviours and goals. Typically, a persona is composed from real data collected from multiple individuals that is translated into a fictional character. Creating personas that range from typical to atypical users helps to understand potential user profiles and facilitates human-centered design.

Personas are a rich representation of users that go beyond mere demographic details and that provide insight into attitudes and behaviours. They furthermore serve to build empathy among the designers toward the user and make the user types memorable to the design team.

Personas typically contain information on

- Demographic details, e. g. age, income, profession
- Personal details, e. g. short biography, name, picture
- Attitudinal and/ or cognitive details, e. g. the persona's mental mode and pain points
- Goals and motivations for using the product
- Behavioural details in situations with the product

In Reflow, personas were collected during two workshops with the pilots held in M9/M10. However, requirements will naturally change and evolve during the project. As we adopted the agile methodology, new

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

personas can be written and added every iteration. The personas created by the pilots Paris, Cluj, Vejle and Amsterdam can be found in the appendix in Section 6.3.

2.3 User Stories elicitation

A user story is a tool often used in agile software development, describing a feature of the software from the end user's perspective. As user stories clearly state the different type of users, what they desire and why, they can be seen as a very high-level definition of requirements.

A user story is short and consists generally of one sentence only. Still, user stories contain enough information about the requirement so that developers can work with them and estimate the effort needed to implement the feature.

A user story typically follows a simple template:
When I < situation>, I want to <user-requirement> so that <reason>.

In Reflow, user stories were collected during two workshops with the pilots held in M9/M10. However, requirements will naturally change and evolve during the project. As we adopted the agile methodology, new user stories can be written and added every iteration.
The template for the collection of the user stories is shown in Table 2.

Table 2: Reflow User Stories Template with some example user stories.

Id	Theme	User story			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
A01	Material Tracking	order yarn	know the composition	guarantee the quality	High	ReflowOS allows the user to trace back the processing of a respective material within the Reflow system
P04	Search	Want to design furniture	know where I can find close materials	Adjust my small batch product	High	ReflowOS must allow users to see the whole processing chain of a certain material

The template is structured as follows:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

- ID: a unique ID consisting of the initial letter of the pilot (A=Amsterdam, B=Berlin, C=Cluj, M=Milan, P=Paris, V=Vejle) and a consecutive number.
- THEME: the theme groups user stories belonging to the same topic. Themes could be e.g. Query Tool, Browse services, Notifications, Data Management etc. The themes are a useful way to think of different functionalities of ReflowOS.
- USER STORY:
    - *When I < situation>:* defines the situation in which the requirement occurs.
    - *I want to <user requirement>:* describes the functionality that is to be added to ReflowOS. This field is free.
    - *So that I can <reason>:* describes the added value of the development suggested. This field is free.
- VALUE: indicates the added value (high, medium, low) to the ReflowOS solution from the defined story (from pilot perspective).
- ACCEPTANCE: the conditions of satisfaction of the story.

The user stories were created in workshops between the technical partners and the pilots. For this, the pilots were separated into two groups based on their main interest in ReflowOS: Material Exchange (Group A: Amsterdam, Paris, Vejle) or Open Data Dashboard (Group B: Cluj Napoca, Milan,). For each group, a workshop was conducted in order to identify the user stories. The workshops took place in February / March 2020. The Berlin pilot did not take part in any of the workshops, as at that point the pilot was not finally constituted. The Milan pilot was not able to take part in the workshop due to travel conflicts. All user stories collected are presented in the Appendix, Section 6.1.

2.4 Use Cases

Use cases describe the interactions between the user and a software system. In contrast to a business process, use cases are more specific and dialled in, as they describe on granular level how the user interacts with the system. Use cases not only describe the basic flow of a user interacting with a system, but also considers exception handling and alternative flows. Thus, use case are a great communication tool, describing in a non-technical way how the user should be able to interact with the system and how the system should react to that. In general, this creates a more concrete vision of how the end product should behave and also helps the developers implementing the software.

For the description of use cases for ReflowOS and the Open Data Dashboard, the following template was used. The use cases can be found in the Appendix in Section 6.4.

Table 3: Use Case Template.

Name	*Name in form of <action> + <object>*
ID	*Identifier unique to each use case*
Goal	*The goal in context of the use case as a short active verb phrase*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Brief description	*Brief sentence to describe the use case. E.g. "Starts when".."Ends when"*	
Actors	*Type of user / role who interacts with the system*	
Preconditions	*Any state the system must be in or condition that must be met before the Use case is started*	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	*put here the steps of the scenario from start (trigger) to goal delivery, and any clean-up after*
	2	*…*
Alternate flows	*Alternate path through the system*	
	Step	*Alternate Action*
	1	*<condition causing branching>: action or name of sub-use case*
	2	*…*
Exception flows	*Exception handling by the system* *<condition for the exception to be called>*	
Post-conditions	*Any state the system must be in or condition that must be met after the Use case is completed successfully. E.g. information that needs to be stored, output that needs to be generated*	

2.5 User Requirements Definition

The user stories were used to extract the user requirements. Once the user stories have been collected in the workshops, the technical partners retrieved the requirements. Requirements that describe functionalities which should be available for all pilots are classified as "core requirements".

Requirements that might be of interested only for one pilot, are referred to as "pilot Requirements". The template for the requirements is shown in Table 4.

Table 4: Requirement Definition Template.

ID	Description	References	Core (CR) / Pilot (PR) Requirement	Priority
FR03	ReflowOS should allow marking offers as temporary	P05	CR	Low
FR28	The Open Data Dashboard should be able to filter the search results	C04	CR	High

The Requirement Excel sheet is structured as follows:

- ID: an arbitrary ID
- DESCRIPTION: the description of the requirement
- REFERENCES: the user story ID and the Use Case ID the requirement comes from

- CORE (CR) / PILOT (PR) REQUIREMENT: the technical partners indicate if the requirement is a core or a pilot requirement
- PRIORITY: the technical partners assess the priority of the requirement. The priority can be set as low, medium, high or very high

2.6 Next iteration planning

As already mentioned in Section 2.1, we follow an iterative approach when developing both, ReflowOS and the Open Data Dashboard. As both tools are developed independently, the iteration planning is separated. Table 5 shows what has been done during the first iterations and what will be done in the upcoming iterations for ReflowOS, Table 6 does the same for the Open Data Dashboard.

Table 5: Iteration Planning for ReflowOS.

Iteration	Months	Actions
1	M6-M11	Use Cases and scenarios for ReflowOS
2	M9	User stories definition for Amsterdam, Paris and Vejle (workshop)
3	M10	Presentation of workshop results to the rest of the pilots, collecting feedback.
4	M12	First test of the REFLOWOS MVP to collect initial feedback
5	M12-14	UI design guidelines development. These are to be shared with technical partners
6	M16-19	REFLOWOS front end development (coordinated with all partners from WP2)
7	M20-26	Iteration with pilots to support them in their API customization (Format to be defined: workshops, meetings, etc.)

Table 6: Iteration Planning for the Open Data Dashboard.

Iteration	Months	Actions
1	M6 – M9	User stories and use cases for Cluj (workshop) and Berlin
2	M10	Presentation of workshop results to the rest of the pilots, collecting feedback
3	M11	Presentation of mock-ups to pilots and collection of feedback
4	M12	Development infrastructure is set up
5	M14	First operational version of an Open Data Dashboard is deployed and can be used by pilots
6	M20	Second operational version of an Open Data Dashboard is deployed and can be used by pilots
7	M24	Submission of D2.4
8	M25 - 36	Maintenance

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

3. Scenarios and Use Cases

3.1 General Scenarios

This section introduces three general scenarios that show how pilots can use ReflowOS in the future. These scenarios have been developed on the basis of initial talks with each pilot, and in order to give the pilots more guidance and a vision of what will be possible with ReflowOS. These general scenarios have later been adjusted to the respective pilots needs in form of pilot scenarios, which can be found in Section 3.2. The pilot scenarios were derived from the workshop results in Amsterdam and Cluj.

3.1.1 Scenarios for ReflowOS

Resource Flow: Track and Trace Scenario (Economic Foundation)

This scenario can be seen as the economic foundation, as this is the base scenario, which the other two scenarios need to build on.
Goals:

- Backtracking of supply chains
- Locate resources and skills
- Offers/needs feed

Pilot participants can log the transfers of a material from one place to another or any kind of economic activities that has been performed on a specific resource. Based on each pilot policy, other stakeholders can locate where those materials are physically stored and which activities has been done in order to produce a good. Such economic activities may or may not be facilitated by digital agreements. Goods that are produced by supply chains within the reflow pilot can be later identified by digital codes (e.g. QR codes or barcodes), so that participants can trace back the "trip" of that material, with each material receiving a kind of material passport.
That could be enhanced and interrelated with additional metadata, such as resource consumption, or material transformations, in order to obtain focused data and build ad-hoc narratives for specific supply chains. Transfers can occur within an organization or between different stakeholders in a safe and secure way.

Observatory of real-time circular economy activities

Goals:

- Promote and raise consciousness on citizenship around CE activities that are in place in the pilot urban and peri-urban area
- Affect upcoming policies

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Those goals can be reached through data visualisations and ad-hoc reports generated from different sources and accessible by their own chosen audience. The observatory software will enable stakeholders to insert manually or automatically (through sensors / scanning / etc.) data about the material flows they want to track, at custom level of details.

Data visualisations will be far from being static images, but will mutate and update constantly based on live data. In addition, users can customize data visualizations charts styles accordingly to which metrics they want to highlight and monitor from the dataset produced.

3.1.2 Scenario for Open Data Dashboard

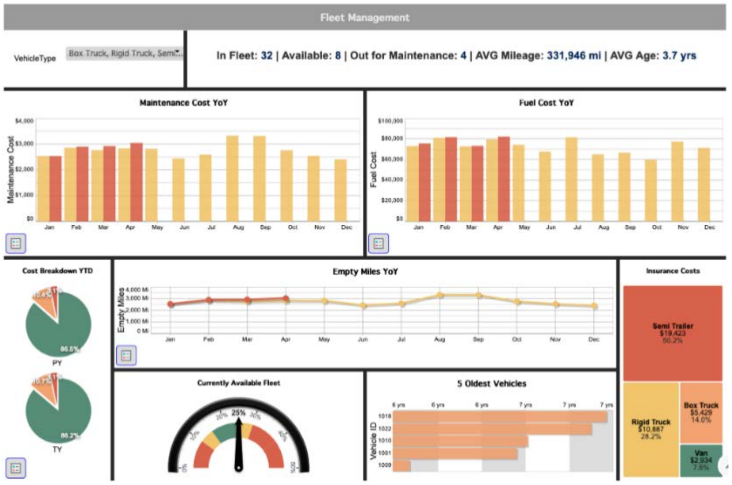
Open Data Dashboard / Dashboard

Goals:

- Provide access to open data circulated/generated in REFLOW that can be used both by humans (GUI) and machines (API)
- Provide a possibility to collect relevant data from other sources or publish own data
- Visualise data collected in the Reflow Open Data Dashboard
- Combine visualisations in order to create meaningful dashboards

Reflow project and especially its pilots applications will generate a significant amount of useful data, which partially can be published as open data and be reused by the community. Examples of these data are the description of the materials, public information about material flows, information about the involved organizations, waste heat potentials, etc. The Open Data Dashboard will provide a possibility to find published datasets, publish own data, register important datasets already published by third parties, download data or visualize data. In addition, created visualisations can be published and share with others. These published visualisations can be combined in a dashboard view. Figure 3 shows a screenshot of an example dashboard, taken from https://www.idashboards.com.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Fleet Management

VehicleType Box Truck, Rigid Truck, Semi...

In Fleet: 32 | Available: 8 | Out for Maintenance: 4 | AVG Mileage: 331,946 mi | AVG Age: 3.7 yrs

Figure 3: An example dashboard, showing information about fleet management (taken from https://www.idashboards.com).

3.2 Pilot Scenarios

The general scenarios introduced in Section 3.1 were created for the pilots to see what ReflowOS in general could be used for. Once the pilot had decided which scenario would suit their local situation best, the respective general scenario was transformed into a pilot specific scenario by adjusting it to the corresponding local ecosystem. This was done built on the results from the workshops in Amsterdam and Cluj. In the following, the pilot specific scenarios for Amsterdam, Paris, Berlin and Cluj are presented.

3.2.1 Amsterdam Scenario

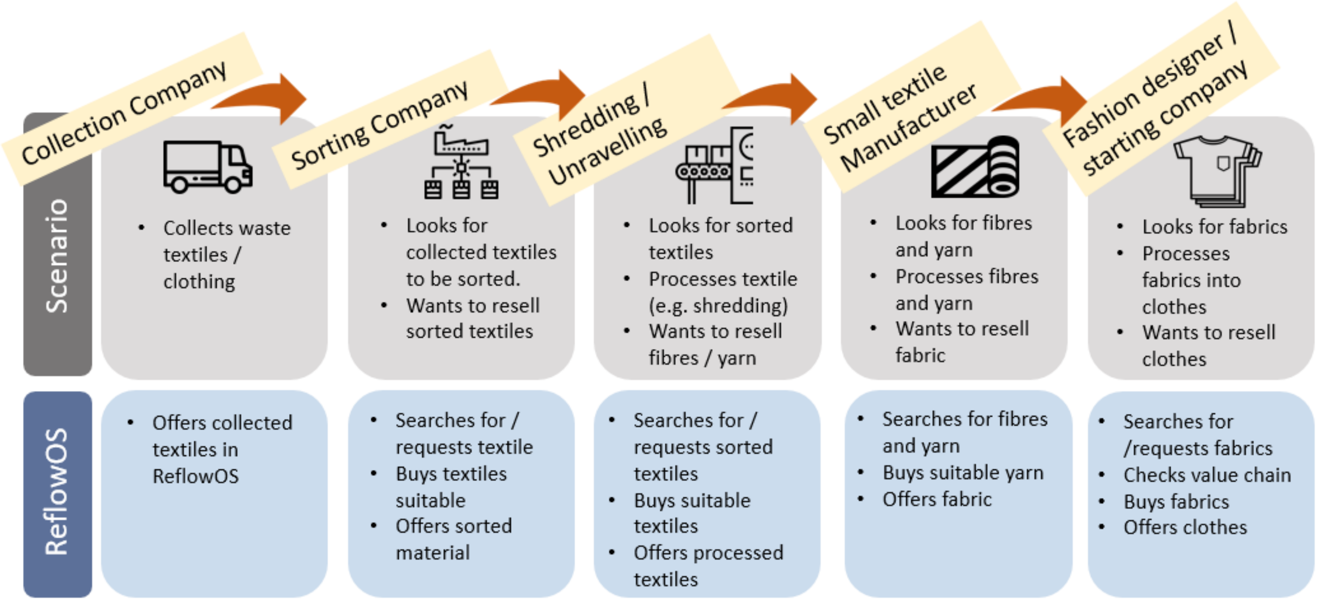
The pilot's general topic is the lifecycle of discarded textiles. The aim is to bring back textile waste into the material stream and increase the collection of clothing and home textiles that can be recycled to be made circular. Different actors can be identified along the way of recycling textiles. ReflowOS can help to network these actors, enable material exchange amongst them and thus map the entire value chain. In concrete terms, an Amsterdam scenario for the use of ReflowOS could look as follows:

The collection company that collects textiles and clothing from e.g. street bins create offers in ReflowOS selling the material by the container. Sorting companies are interested in unsorted material and find the offer from the collection company in ReflowOS. Alternatively, they put requests for unsorted textile in ReflowOS. The material is passed on from the collection company to the sorting company via ReflowOS.

The sorting company sorts the textiles and creates new offers for selling the sorted material in ReflowOS. In the next step, companies which are specialized in shredding and unravelling textiles accept the offer from the sorting company and buy the textile by kg. After processing the textiles, they offer fibres and yarn in ReflowOS. Small textile manufacturers can find these offers in ReflowOS, buy the material and process it to fabric. This fabric again can be offered in ReflowOS. Fashion designer and starting companies that are

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

interested in creating sustainable clothes can easily check the processing steps of the material in ReflowOS and thus, buy fabric that suits their needs.



	Collection Company	Sorting Company	Shredding / Unravelling	Small textile Manufacturer	Fashion designer / starting company
Scenario	• Collects waste textiles / clothing	• Looks for collected textiles to be sorted. • Wants to resell sorted textiles	• Looks for sorted textiles • Processes textile (e.g. shredding) • Wants to resell fibres / yarn	• Looks for fibres and yarn • Processes fibres and yarn • Wants to resell fabric	• Looks for fabrics • Processes fabrics into clothes • Wants to resell clothes
ReflowOS	• Offers collected textiles in ReflowOS	• Searches for / requests textile • Buys textiles suitable • Offers sorted material	• Searches for / requests sorted textiles • Buys suitable textiles • Offers processed textiles	• Searches for fibres and yarn • Buys suitable yarn • Offers fabric	• Searches for /requests fabrics • Checks value chain • Buys fabrics • Offers clothes

Figure 4: Amsterdam Scenario for ReflowOS.

3.2.2 Paris Scenario

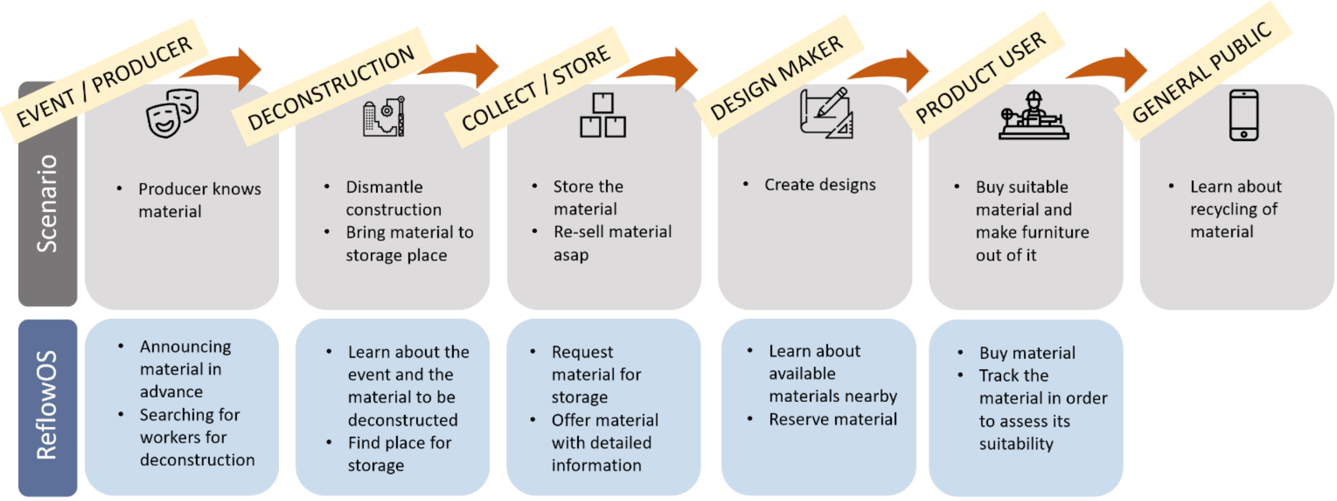
The pilot's general topic is to build a CE approach for event management and temporary structures aiming to coordinate the use and reuse of materials (primarily timber) between events and structures in the territory of Great Paris. Many different actors can be involved in the material flow of the timber. ReflowOS can help to network these actors, enable material exchange amongst them and thus map the entire value chain. In concrete terms, a Parisian scenario for the use of ReflowOS could look as follows:

An event organizer knows what kind of material is being used for the event construction and which part of it he wants to pass on for reuse. He then puts an announcement into ReflowOS that the respective material is available after the event and temporary structure finishes. For interested parties to get a better understanding, the organizer describes the material as detailed as possible adding metadata about the material characteristics, and uploads images of it into ReflowOS. He also uses ReflowOS to search for workers that will perform the deconstruction of the material as soon as the event is over. The deconstruction worker learns about the event and the request from the organizer via ReflowOS. After agreeing on a contract, the worker performs the deconstruction. The construction consultant (or respectively the event organizer) uses ReflowOS to search for a storage place for the deconstructed material. In ReflowOS he finds an announcement of a company requesting material for storage. The material is brought to the storage place and the storage owner has to re-sell it as soon as possible to

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

keep his costs low. Thus, he offers the material with detailed descriptions and including the geometry in ReflowOS. A design maker finds the material offer on ReflowOS and starts creating a design for the respective material. He also reserves the material via ReflowOS to make sure, that it is still available after finishing his designs. He then creates a new offer in ReflowOS for the material and the design he did. In the next step a product user sees this offer and buys both, the material and the design from the design maker and makes a furniture out of it. All steps which have been undertaken from the time the material was initially entered into ReflowOS by the event organizer are tracked in ReflowOS via QR-Code. The end-user of the furniture can easily access this documentation of the value chain in ReflowOS via the QR code and thus learns about the recycling of the material.

The scenario is presented in Figure 5.



	EVENT / PRODUCER	DECONSTRUCTION	COLLECT / STORE	DESIGN MAKER	PRODUCT USER	GENERAL PUBLIC
Scenario	• Producer knows material	• Dismantle construction • Bring material to storage place	• Store the material • Re-sell material asap	• Create designs	• Buy suitable material and make furniture out of it	• Learn about recycling of material
ReflowOS	• Announcing material in advance • Searching for workers for deconstruction	• Learn about the event and the material to be deconstructed • Find place for storage	• Request material for storage • Offer material with detailed information	• Learn about available materials nearby • Reserve material	• Buy material • Track the material in order to assess its suitability	

Figure 5: Paris Scenario for ReflowOS.

3.2.3 Cluj Scenario

The Rumanian National Energetic Strategy for 2030 sets objectives such as energy security, sustainable development and competitiveness that also apply for the City of Cluj-Napoca. Therefore, the city is interested in a tool that helps e.g. proving how the measures taken to date by the City have impacted energy efficiency and thus showing that it makes sense to invest in energy efficiency. The information gathered should not only be disseminated throughout potential investors but also to the broad public in order to acquire a behavioural change of consumers. For this, the Reflow Open Data Dashboard could be used.

The energy department of the Municipality of Cluj-Napoca relies on data from the energy provider for writing reports. Currently, these data are requested manually from the energy provider by an employee of the energy department. This process poses several problems: It often takes a long time until the requests are processed

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

and data is delivered, sometimes data is missing, data delivered is diverse and needs manual effort to be processed.

With the usage of the Reflow Open Data Dashboard, data can be exchanged automatically between the energy provider and the Municipality. The Dashboard provides an API for the automated upload of the data. This might make the manual request of data and the time-consuming communication between the municipality and the energy provider obsolete. Active users of the Open Data Dashboard, such as the energy department of the Municipality, can search the Dashboard for available data, select data, select a type of visualisation from a given standard set, create a data visualisation and publish it. Published visualisations can be aggregated into a dashboard. Passive users (users that do not create visualisations) can search for data they are interested and look at visualisations. Thus, the Open Data Dashboard can provide useful information on energy consumption or public spending for citizen. The user types and their actions regarding the Open Data Dashboard are depicted in Figure 6.

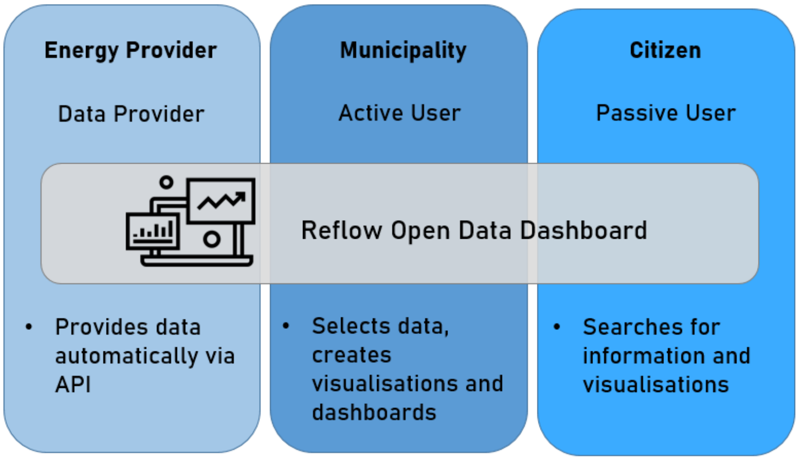

Energy Provider
Data Provider
Municipality
Active User
Citizen
Passive User
Reflow Open Data Dashboard
• Provides data automatically via API
• Selects data, creates visualisations and dashboards
• Searches for information and visualisations

Figure 6: Cluj Scenario for the Open Data Dashboard.

3.2.4 Berlin Scenario

A fundamental transformation of the heat supply with a significant increase in the share of renewable energy sources is necessary to reach the Berlin climate targets. In line with the Berlin Energy and Climate Protection Program 2030, the REFLOW project Pilot Berlin aims to reduce CO2 emissions and increase energy efficiency in the German capital. Specifically, the project aims to implement a digital and organizational platform for optimizing the use of wastewater-heat. Wastewater-heat, both from residential and commercial sources, is a particularly relevant category of waste heat in the urban context, as the mostly underground main water pipes cover large parts of the city and therefore allow for an extensive distribution of the energy used. This was a decisive factor in the selection of the Berlin pilot's focus. By implementing different measures, wastewater-heat potentials will be identified, and the visibility of the technology will be increased. At the core of these efforts is the idea of coupling useful data from the supply side with data from the demand

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

side and even with data from complementary heat generation systems. The result is a reliable picture of where and how it is possible to heat in Berlin according to CE standards. For this, the Reflow Open Data dashboard could be used.

Dashboard Use

Fundamental data concerning wastewater heat potentials could be provided to the platform by Berliner Wasserbetriebe (BWB). As things stand today, it is possible to couple these data with several other datasets such as: district heating grid, public buildings, new building projects (including contact points) and heat demand. Where legally possible, the Berlin Pilot's private organizations AH and MCS will be responsible for feeding in those supplementary data sets. The datasets and coupling results should be shown on a map in a fashion that is also comprehensible to those not familiar with the subject, but without reducing the information value. This map could be integrated into stakeholder workshops, neighbourhood development processes or, more generally, to raise public awareness for wastewater heat technology, which up to now has led a shadow existence.

The story

A real estate developer wants to build or modernize one of her buildings. Due to new building regulations, but also because being "sustainable" is the "new normal" in the real estate industry, she wants to use renewable energy sources for the new heating system. She learns about an interesting new application for identifying CE applications when talking to her point of contact from the Neukölln district building authority, which works closely with and was briefed by the Berlin Pilot consortium: ReflowOS. This gives her the opportunity to check if there is potential for the use of wastewater as a heat source at the location of her building. At the Open Data Dashboard she sees the amount of power that can be extracted from the sewage system nearby. If the evaluation indicates a higher potential than she is able to use with her own new building, the platform offers the opportunity to check if there is another possible heat consuming party that has registered a heat demand in the ReflowOS map. If that's the case both stakeholders can connect and build a wastewater heating station together. However, it is not unlikely that even both parties in a joint venture would shy away from the effort and expense of installing and financing a wastewater plant. They might also know too few best practices from their colleagues. In the future, the Berlin Pilot may be able to plan, finance, install and operate wastewater-heat systems through an umbrella association (ger.: Dachgenossenschaft) for wastewater-heat at low cost and possibly with the help of government subsidies. In this case ReflowOS would be an excellent filter for the identification of possible customers (ger: Genossen). All this leads to an efficient exploitation of the wastewater heat potential, making Berlin future-proof in terms of heat supply.

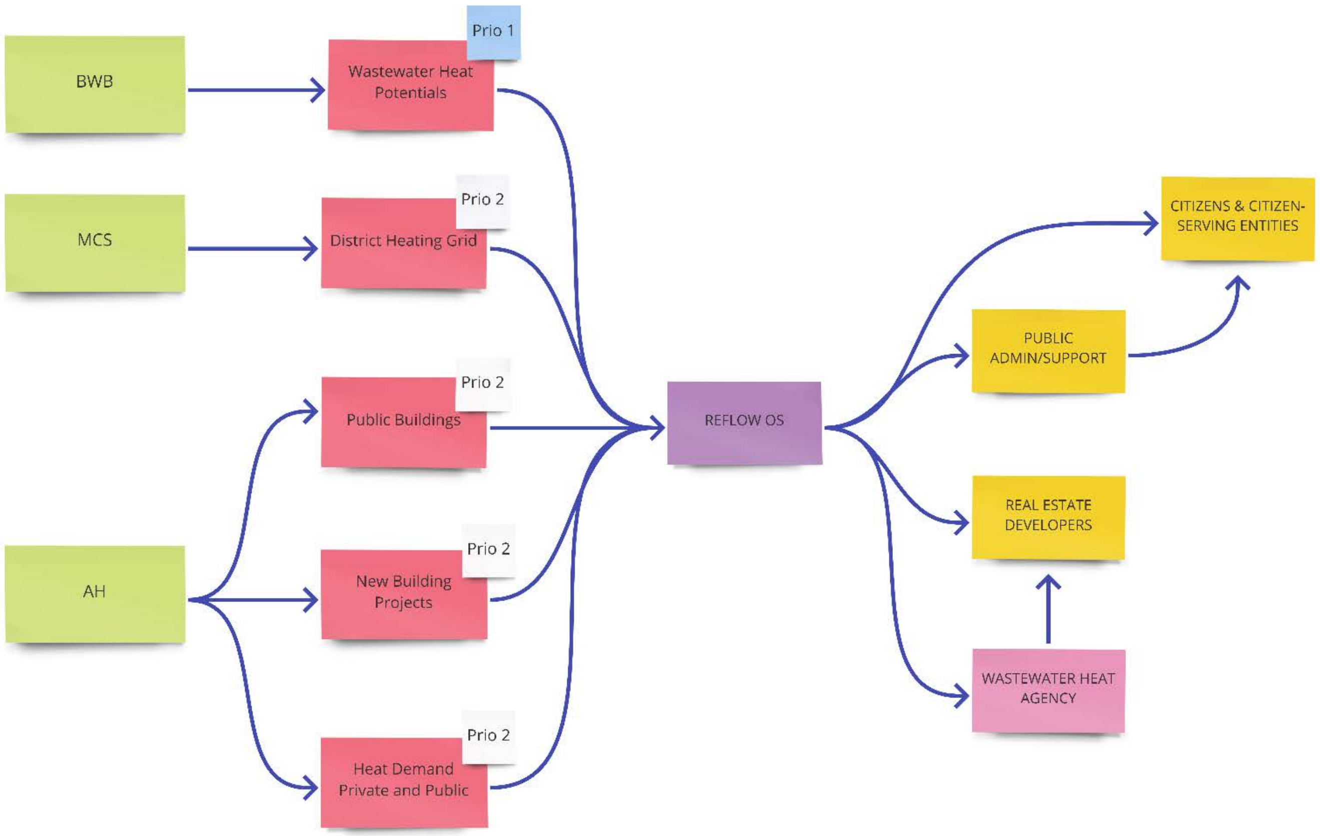


Figure 7: Berlin Pilot Scenario and stakeholders.

3.3 Identified Actions

ReflowOS will allow users to perform various actions. Table 7 shows an overview of the planed actions users can perform in ReflowOS to be implemented over the project runtime. As the Open Data Dashboard does not have a material flow like ReflowOS, it is not mentioned in the table below.

Table 7: ReflowOS actions.

Action	Description
Request	Request a resource or a skill from the network
Offer	Offer a resource or a skill to the network
Transfer	Transfer a material from one person to another (one-way).
Work	An individual or a robot puts some effort inside the process. The action is not strictly related to a resource. (considered as input)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Use	The resource is used, but not consumed. After usage, it is still there in same quantity (considered as input)
Consume	After the consumption of the resource, the quantity is affected (considered as input)
Produce	A new resource is created (considered as output)

4. User Requirements

As mentioned in Section 2.5 the user requirements were extracted from the user stories and have been analysed and validated by the technical partners.

The requirements were split into two main features:

- Functional or non-functional requirement
- Core or pilot requirement

Table 8 reports an overview of the main features of each characteristic:

Table 8: Requirement characteristics.

Characteristic	Description
Functional Requirement	Functional requirements define the required behaviour of the system to be built. They define the capabilities that a product must provide to its users. Functional requirements are based on system objectives and respond to the critical task of ensuring the right implementation of the expected functionality in the final software.
Non-Functional Requirement	Non-functional requirements specify other characteristics of ReflowOS, other than functionality. These requirements can be subcategorized into categories such as performance, design constraints, etc. Non-functional requirements can also describe quality attributes or implementation constraints that the product must have. Thus, they are more qualitative than functional requirements.
Core Requirement	Core requirements are the requirements for functionalities that ReflowOS and the Open Data Dashboard should offer and will be available for each pilot.
Pilot Requirement	Pilot requirements are pilot specific requirements that are only of interest for a selection of pilots

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

4.1 Functional Requirements

Table 9 shows the collection of functional requirements.

Table 9: Functional Requirements.

ID	Description	References	Core (CR) / Pilot (PR) Requirement	Priority
ReflowOS				
Material Offering				
FR01	ReflowOS allows users to publish an offer of an available resource or skill	US:P07, UC:ROS07	CR	Medium
FR02	ReflowOS should allow the seller of material to combine different orders	US:A03	PR	Low
FR03	ReflowOS should allow marking offers as temporary	US:P05	CR	Low
FR04	ReflowOS should allow the creator of an offer to change the respective offer if necessary		CR	Medium
FR05	ReflowOS allows users to start a thread on an existing offer or request	UC:ROS09	CR	Medium
Material Tracking				
FR06	ReflowOS should be able to display the processing chain of a respective material or resource from the point the material entered the Reflow system	US:A01, US:A02, US:A05, US:A06, US:A08, US:A09, US:A12, US:A19, US:P03, US:P08, US:P13, UC:ROS13	CR	High
FR07	ReflowOS allows user to track the consumption of a new resource	UC:ROS16	CR	High
FR08	ReflowOS allows users to track the production of a new resource	UC:ROS16	CR	High
FR09	ReflowOS allows users to track the transfer of a resource to another agent	UC:ROS15	CR	High
FR10	ReflowOS allows users to track the usage of a resource	UC:ROS16	CR	High
Search				
FR10	ReflowOS has to provide a search function for material and resources	US:A07, US:A15, US:P02, US:P10, UC:ROS11	CR	Medium
FR11	ReflowOS should offer a geo-based search for material and resources	US:P01, US:P11	CR	Medium
FR12	ReflowOS has to provide filter options to refine the search results		CR	Medium
Social Network				
FR13	ReflowOS allows users to follow and unfollow other users	UC:ROS04, UC:ROS05	CR	High
FR14	ReflowOS should allow the discovery of other users that are part of the network	UC:ROS03	CR	High

FR15	ReflowOS allows users to edit their profile information	UC:ROS02	CR	High
Data Input				
FR16	ReflowOS should provide a set of pilot specific metadata fields for the seller to describe the material offered	US:A10, US:P06, US:P12	CR	Medium
FR17	ReflowOS should allow uploading pictures of the material offered	US:A11	CR	Medium
FR18	ReflowOS should provide capability to manage data sources (import, select, upload, download) from/to different formats		CR	Low
FR19	ReflowOS should be able to process sensor data		CR	Medium
FR20	ReflowOS should be able to process positioning information and location data	US:PO1	CR	Low
Security				
FR21	ReflowOS should implement end-to-end encryption		CR	High
FR22	ReflowOS should allow user to choose between different level of visibility for their economic activities		CR	Medium
Others				
FR23	ReflowOS allows users to create requests for material and resources	US:A14, UC:ROS08	CR	Medium
FR24	ReflowOS provides data visualisations functions	US:A04	PR	Low
FR25	ReflowOS allows users to reserve material for a certain period of time	US:P11, UC:ROS12		Low
FR26	ReflowOS should be able to link all changes that happen to a resource to a user		CR	Medium
FR27	ReflowOS should allow the creation of different users/groups and access rights for authorized system user		CR	High
FR28	ReflowOS allows users to create custom taxonomies	UC:ROS06	CR	High
FR29	ReflowOS allows users to start a thread on an offer or request	UC:ROS09	CR	Medium
FR30	ReflowOS allows users to subscribe to taxonomies	UC:ROS10	CR	High
Open Data Dashboard				
Search				
FR31	The Open Data Dashboard has to provide a search function for metadata	US:C01, US:C03, US:C04 US:C05, UC:ODD01	CR	High
FR32	The Open Data Dashboard should be able to filter the search results	US:C04, UC:ODD06	CR	High
FR33	The Open Data Dashboard provides a geo-based search on a map	US:C15	CR	Medium
Data Visualisation				
FR34	The Open Data Dashboard should be able to process real-time data		CR	Low
FR35	The Open Data Dashboard should be able to process static data		CR	High
FR36	The Open Data Dashboard should allow uploading of users' data to be processed and visualised	US:C06	CR	High

FR37	The Open Data Dashboard should allow to select data sources and type of visualisation	US:C14, UC:ODD04	CR	High
FR38	The Open Data Dashboard has to support different standard visualisation charts	US:C14, UC:ODD04	CR	High
FR39	The Open Data Dashboard should allow the publication of visualizations		CR	High
FR40	The Open Data Dashboard should allow the aggregation of visualisations to a dashboard	US:C08, UC:ODD05	CR	High
FR41	The Open Data Dashboard should allow the arrangement of visualisations within the dashboard per drag and drop		CR	High
FR42	The Open Data Dashboard should allow the publication of dashboards	US:C08	CR	High
Data input				
FR43	The Open Data Dashboard should provide capability to manage data sources (import, select, upload, download) from/to different formats	UC:ODD03	CR	High
FR44	The Open Data Dashboard should be able to process positioning information and location data	US:C15	CR	Medium
FR45	The Open Data Dashboard should be able harvest data from other Open Data Dashboards	US:CO9	CR	High
Security				
FR46	The Open Data Dashboard should be able to anonymise personal data		PR	Low
FR47	The Open Data Dashboard should allow the creation of different users/groups and access rights for authorized system user		CR	Low
Others				
FR48	The Open Data Dashboard should be able to perform quality check regarding the completeness of data imported or uploaded	US:C11	PR	Low
FR49	The Open Data Dashboard should support combinations of datasets from different sources	US:C12	CR	Medium

4.2 Non-Functional Requirements

To identify the non-functional requirements, the model proposed by ISO/IEC 25010:2011 was adopted. Following that model there are eight quality characteristics contributing to software product quality: Functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability and portability. For each of these categories, ISO/IEC 205010:2011 lists several sub-categories. The sub-categories help to get a better understanding of the categories and are thus listed in Table 10.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Table 10: ISO/IEC 25010:2011 Software Product Quality Model incl. Sub-Categories.

Quality Characteristic	Sub-categories
Functional Suitability	• Functional Completeness • Functional Correctness • Functional Appropriateness
Performance efficiency	• Time Behaviour • Resource Utilization • Capacity
Compatibility	• Co-existence • Interoperability
Usability	• Appropriateness Recognisability • Learnability • Operability • User Error Protection • User Interface Aesthetics • Accessibility
Reliability	• Maturity • Availability • Fault Tolerance • Recoverability
Security	• Confidentiality • Integrity • Non-repudiation • Authenticity • Accountability
Maintainability	• Modularity • Reusability • Analysability • Modifiability • Testability
Portability	• Adaptability • Installability

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

	• Replaceability

Table 11 shows the collection of non-functional requirements for ReflowOS, Table 12 shows the non-functional requirements for the Open Data Dashboard.

Table 11: Non-functional requirements ReflowOS.

Requirement Sub-category	Id	Description
Functional Suitability	NFR1	ReflowOS should respect all high priority functional requirements.
Performance efficiency	NFR2	ReflowOS should response to any user request within two seconds.
Compatibility	NFR3	ReflowOS should support major desktop and mobile web browsers (Chrome, Safari and Firefox)[1]
Usability	NFR4	ReflowOS should have a multi-language user interface
	NFR5	ReflowOS shall feature a user-friendly interface, provide an overview of supported actions through a user guide / helpbox
Reliability	NFR6	ReflowOS core modules have to be properly tested and documented
Security	NFR7	ReflowOS should take into account privacy and security rules
Maintainability	NFR8	ReflowOS should be easily maintainable
	NFR9	ReflowOS should be able to raise alarms about hardware/software failures of the solution
Portability	NFR10	ReflowOS should be able to be deployed in a timely and efficient manner

[1] "Browser Market Share Worldwide – StatCounter Global Stats". StatCounter Global Stats.

Table 12: Non-functional requirements Open Data Dashboard

Requirement Sub-category	Id	Description
Functional Suitability	NFR11	The Open Data Dashboard should respect all high priority functional requirements.
Performance efficiency	NFR12	The Open Data Dashboard should response to any user request within two seconds.
Compatibility	NFR13	The Open Data Dashboard should support major desktop and mobile web browsers (Chrome, Safari and Firefox.
Usability	NFR14	The Open Data Dashboard should have a multi-language user interface
	NFR15	The Open Data Dashboard shall feature a user-friendly interface, provide an overview of supported actions through a user guide / help box.
Reliability	NFR16	The Open Data Dashboard should stay in a reliable state, even if failures occur.
Security	NFR17	The Open Data Dashboard should offer login with user credentials
	NFR18	The Open Data Dashboard should take into account privacy and security rules
Maintainability	NFR19	The Open Data Dashboard should be easily maintainable
	NFR20	The Open Data Dashboard should be able to raise alarms about hardware/software failures of the solution
Portability	NFR21	The Open Data Dashboard should be able to be deployed in a timely and efficient manner.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

5. Conclusion

The objective of this deliverable was to deliver the use cases and requirements for ReflowOS and the Open Data Dashboard. More specifically, in order to collect and analyse the use cases and requirements, the agile software development extraction techniques were adopted. At first, the actors of ReflowOS and the Open Data Dashboard were identified by creating Personas with the pilots. The Personas were used for the creation of user stories. In total, 52 user stories were created together with the pilots. The user stories were then transformed into use cases. From the use cases, the functional and non-functional requirements were extracted and collected by the technical partners. Finally, the requirements were analysed and validated by the technical partners.

With the listing of Use Cases and user requirements, this document builds the foundation for the further implementation of both, ReflowOS and the Open Data Dashboard. The Deliverable documents the progress made so far and outlines a concrete vision of both tools. As for the development of the tools we have chosen an iterative process, the requirements and the use cases might change over the remaining project runtime. It is also possible, that over the time when the pilots evolve and become more elaborate, new requirements will be found. However, this document is a solid basis representing the progress made and the decisions taken so far in the context of Work Package 2. It not only adds value for the future implementation work by providing a vision of both tools and documenting the first requirements which directly stem from the pilots, but also for other interested cities outside the project by describing in detail the methodology used to create use cases and collect requirements.

.

6. Appendix

6.1 User Stories

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
ReflowOS						
US:A01	Material Tracking	Order yarn	Know the composition	Guarantee the quality	High	ReflowOS allows the user to trace back the processing of a respective material within the Reflow system
US:A02	Material Tracking	Sell my fabric	Let the designer know that my products are recycled, circular and made with sustainable practices	-	High	ReflowOS allows users to see the whole processing chain of a certain material
US:A03	Material Offering	Get more orders that I can combine	I can have more impact	Push the industry towards something better	Low	A user offering material on ReflowOS is able to combine different orders
US:A04	Data Visualization	Go to a seller	Offer possible products	Show analytical reports supporting me	Medium	ReflowOS has a function for data visualization and ad-hoc reports
US:A05	Material Tracking	When someone buys my clothes	I want the buyer to have access to all the story of the material	Improve the design value	High	ReflowOS allows users to see the whole processing chain of a certain material
US:A06	Material Tracking	Communicate about my products	Know the history of the material	Be honest about my products	High	ReflowOS allows users to see the whole processing chain of a certain material
US:A07	Search	Want to connect with manufacturers	Know where to look for them	Expand my network	High	ReflowOS allows users to search for material and resources
US:A08	Material Tracking	Sell a garment	Let the consumer know where I came from	-	High	ReflowOS allows users to see the whole processing chain of a certain material
US:A09	Material Tracking	Buy sustainable textile	Know where it comes from	Let my buyer know and add	High	ReflowOS allows users to see the whole

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
				value to my piece		processing chain of a certain material
US:A10	Material Offering / Data Input	Am looking for a good material	Learn about the technical aspect of it	-	Very high	ReflowOS provides a function that allows users to describe their resources according to different characteristics
US:A11	Material Offering	Receive a material	Take a picture and upload all the specs	Process some material	High	ReflowOS provides a function that allows users to upload pictures for the description of material
US:A12	Material Tracking	Want to become a textile material hut	Know what I sell on detailed level	Further develop my materials business	High	ReflowOS allows users to see the whole processing chain of a certain material
US:A13	n/a	Sort materials	Sell them faster	Start sorting a new batch		*Out of scope of ReflowOS*
US:A14	Material Request	Search for textile / fibers/ clothes	Know who is interested	Open up my market possibilities	Medium	ReflowOS provides a function that allows user to put request or announcements for material they are looking for
US:A15	Search	Am looking for more feedstock	Find new suppliers	Increase my selection of batches for sorting	High	ReflowOS allows users to search for material and resources
US:A16	Material Tracking / Social network	Have a good collection of collected clothes	Know who I am selling it to	-	Medium	ReflowOS allows the user to track the flow of a respective material within the Reflow system
US:A17	Material Tracking	Recycle my textiles / clothing	Know where they went	Feel good about it	Medium	ReflowOS allows the user to track the flow of a respective material within the Reflow system
US:A20	n/a	Look at my business	Improve waste collection	Charge higher prices		*Out of scope of ReflowOS*
US:P01	Search	Want to design furniture	Know where I can find close materials	Adjust my small batch product	High	ReflowOS offers a geo-based search to find materials in a chosen location

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
US:P02	Search	Leave a location	Quickly have info about the storage	Fit the ordered material	Low	*ReflowOS enables users to search for storage places nearby*
US:P03	Material Tracking	Buy a furniture	Know where the material came from	Know the story of it	High	ReflowOS must allow users to see the whole processing chain of a certain material
US:P04	n/a	Read on Twitter about Paris Fair	See how much materials were recycled	Have an opinion on my city sustainability		*Out of scope of ReflowOS*
US:P05	Material Offering	When wood arrives	Sell it in less than 24 hours	So I need less space to store	Medium	In ReflowOS, users can mark their offers as temporary
US:P06	Material Offering / Data input	Collect material	Get this in my website with as much information	Move more material	Very high	ReflowOS provides a function that allows users to describe their resources according to different characteristics
US:P07	Material Offering	When the event finishes	People to buy pieces of materials and food	Make more money	Medium	ReflowOS allows users to sell material and resources
US:P08	Material Tracking	Buy material	Know where they come from	Calculate and certify the carbon footprint for my customers	High	ReflowOS must allow users to see the whole processing chain of a certain material
US:P09	n/a	Start the event	Know all materials	Reuse or sell it at the end		*Out of scope of ReflowOS*
US:P10	Search	Dismantle a construction	Quickly store materials	Do not waste time	Very high	ReflowOS allows users to search for material and resources
US:P11	Search / Material Request	Begin a project / design	Know and reserve materials	Begin to know what is possible to make	High / Medium	ReflowOS allows users to search for specific material offered in a respective location. / ReflowOS provides a function that allows users to reserve material offer a certain period of time
US:P12	Material Offering / Data Input	Make a furniture	Materials which are well finished	So i can save time on the fabrication	Medium	ReflowOS provides a function that allows users to describe their resources according to different characteristics

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
US:P13	Material Tracking	Make furniture	To track it	Get it back knowing its history	High	ReflowOS must allow users to see the whole processing chain of a certain material
US:V01		When plastic waste is disposed	To be able to rely on it being thoroughly sorted	Initiate automated processes for recycling		*Out of scope of ReflowOS / Data Dashboard*
US:V02		When wrapping goods	Use the solution and material that is cheapest while still meeting hygienic standards	Remain as competitive as possible		*Out of scope of ReflowOS / Data Dashboard*
US:V03		Plan optimization of waste management	Want to document a proof of concept for circular reuse	Convince key stakeholders in my organisation about the economic and environmental feasibility of change in our practice		*Out of scope of ReflowOS / Data Dashboard*
US:V04		am tasked with prepping plastic waste for collection	Processes for doing it right to be easy and faster than our current practice	Do well for the environment without having to work overtime		*Out of scope of ReflowOS / Data Dashboard*
Open Data Dashboard						
US:C01	Search	I want to pay the tax	Find out what to spend my money on	I can request other investments	Very high	The Open Data Dashboard allows users to search for information and data
US:C02	UI	Want information about the city's policies	To know where to look for it	I don't get lost	High	The UI of the Open Data Dashboard allows simple access to the search function
US:C03	Search	Look for information on public spending	A simple access to it	I am able to find what I am looking for	Very high	The Open Data Dashboard allows users to search for metadata
US:C04	Search	Look for information on public spending	Search for specific topics	I only get results I am interested in	Very high	Users can search for specific topics within the Open Data Dashboard

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
						and filter the results with a facetted search
US:C05	Search	I make consumption plans	Find out the previous consumption	Estimate budgets	Very high	The Open Data Dashboard allows users to search for metadata
US:C06	Data Upload	Am asked to provide data	Do it efficiently	I can spend my time on other tasks	High	Data should be automatically uploaded to the Open Data Dashboard
US:C07	Data Management	Buy energy	Have centralized data	I can organize the auction		*Out of scope of ReflowOS / Data Dashboard*
US:C08	Dashboard	Create reports	Want to publish them as a dashboard	Others can easily access and understand it	High	The Open Data Dashboard allows users to publish visualisations as dashboards
US:C09	Data Management	Receive date	To get it in an automated process	I don't need to request it manually	Medium	The Open Data Dashboard provides a harvesting feature for metadata and data
US:C10	Data Management	Receive data	Want it in one standardized format	I can better understand it		*Out of scope of ReflowOS / Data Dashboard*
US:C11	Data Quality	Collect the data I need	Want to see first sight if it is complete ore not	I don't have to spend time on verification		*Out of scope of ReflowOS / Data Dashboard*
US:C12	Visualisation	Have data from different sources	Aggregate and visualize them	I can make nice reports	High	The Open Data Dashboard can combine data and visualise it
US:C013	Search	Am working on my doctoral thesis	Find consumption data	Check the consumption	High	The Open Data Dashboard allows users to search for metadata
US:C14	Visualisation	Look at energy data	See them as visualized data	Better understand it	Medium	The Open Data Dashboard allows users to select data from different sources and to visualize them by selecting from a set of standard visualization charts

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

ID	Theme	User story / Job to be done			Value	Acceptance
		When I <situation>	I want to <user requirement>	So that I can <reason>		
US:C15	Search	Want information on energy consumption	Filter for specific locations	I get information about places I hang out	Medium	The Open Data Dashboard offers a geo-based search to find information related to a chosen location

6.2 Pilot specific characteristics

As shown in Section 3.2 the pilots are focusing on different topics within the Reflow project. Still, the basic implementation of ReflowOS will be the same for every pilot. However, due to the diverse topics, especially the frontend has to be adjusted according to the respective pilot's needs. For example, when a user wants to describe a resource that provides input on the system, he or she needs to fill in metadata. Depending on the pilot's topic, the required metadata changes, e.g. for describing timber one needs other information than for describing textile. In this section, the required metadata needed for describing the resources are listed for pilots that already had such a list when this deliverable was written. However, this is to be understand as first suggestions from the pilot perspective. The importance and inclusion of these specific requirements have to be evaluated later on.

6.2.1 Paris

- ID number based on a nomenclature - could be a mixed between the GPS coordinate, date and the time
- Date of entry
- Localisation of the source (GPS coordinate or adress)
- Name of the site or deconstruction site or building
- Date of the construction if known
- Types of form: paralepidid or free form (only flat)
  - Dimension for paralepidid:
    - x (cm)
    - y (cm)
    - z (cm)
  - Shape for free form :
    - an array of coordinate {x;y} (cm)
    - thickness (cm)
- Volumes (cm3)
- (a way to measure the liability of the measurement and deformation of the piece - this has to be defined)
- Weigh (gr)
- Volume mass (gr/cm3)
- Color
  - R - 0-255
  - G - 0-255

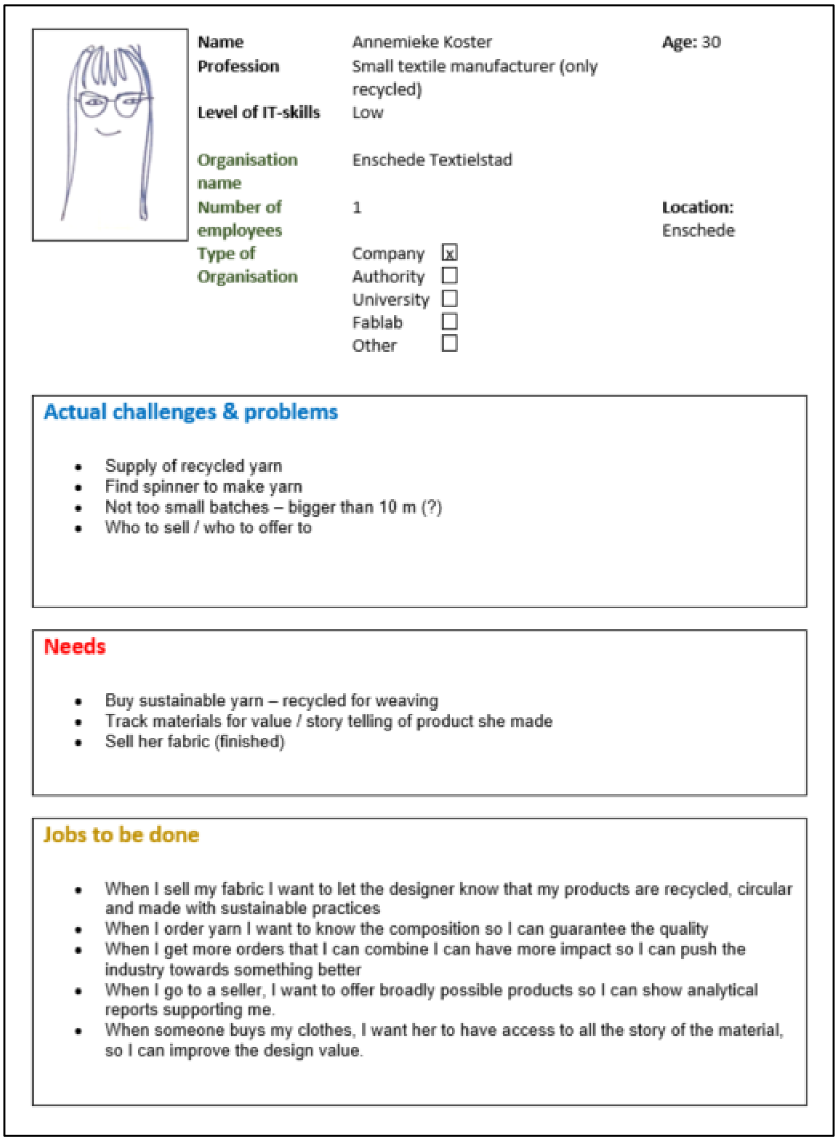
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

  - B - 0-255
- sample of a picture (low res - how to store it?)
- Wood essence
- Store localisation
  - Site where it is store - Name
  - Site where it is store - GPS coordinate
    - Specific nomenclature for the storage (depending of the site - how to be flexible)
- Treatment - Painted / raw / varnish / to be treated
- Lock - possibility to lock some piece in advance for a project
- Date of the lock
- Sold or Used

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

6.3 Personas

6.3.1 Amsterdam



Name	Annemieke Koster	**Age:** 30
Profession	Small textile manufacturer (only recycled)	
Level of IT-skills	Low	
Organisation name	Enschede Textielstad	
Number of employees	1	**Location:** Enschede
Type of Organisation	Company ☒ Authority □ University □ Fablab □ Other □	

Actual challenges & problems

- Supply of recycled yarn
- Find spinner to make yarn
- Not too small batches – bigger than 10 m (?)
- Who to sell / who to offer to

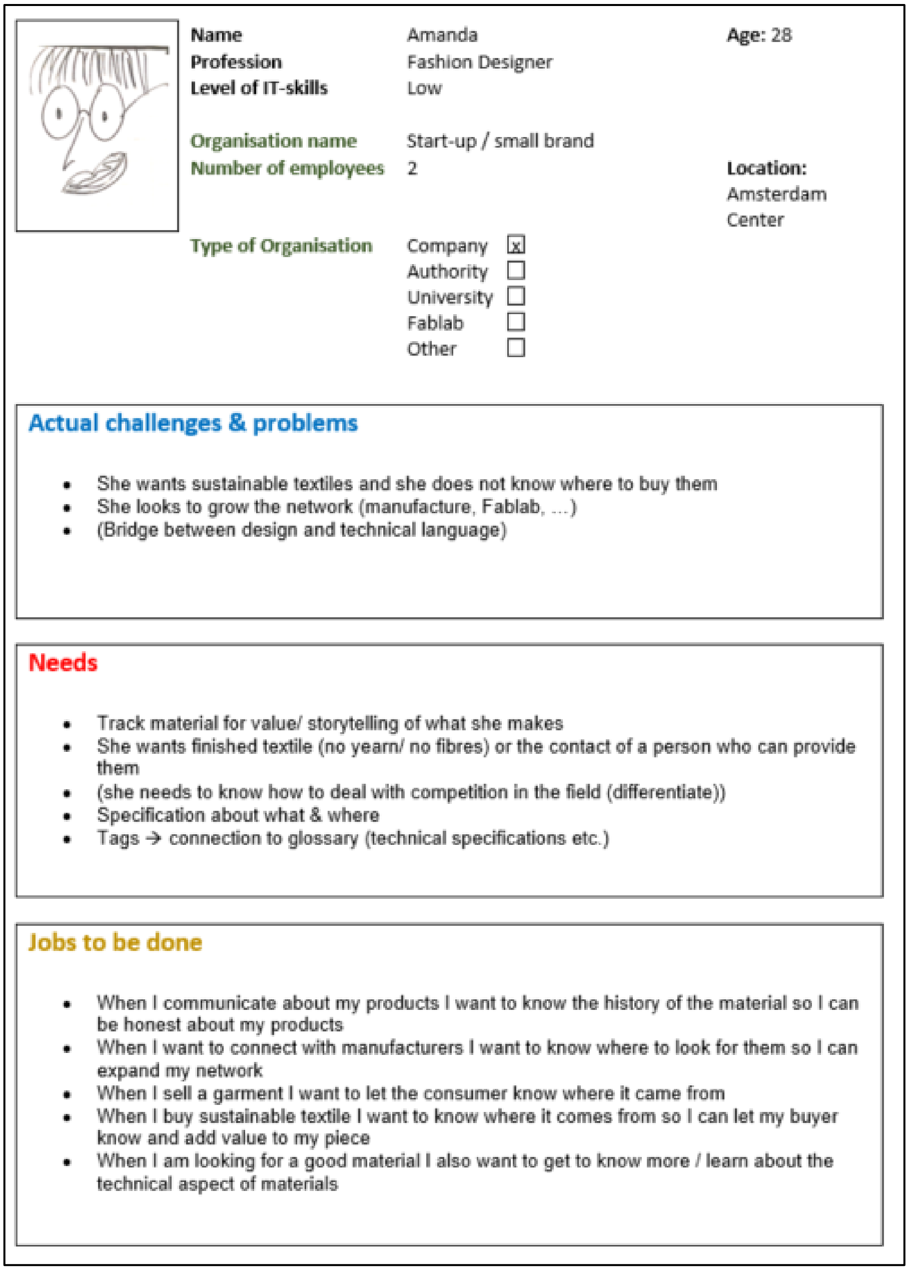
Needs

- Buy sustainable yarn – recycled for weaving
- Track materials for value / story telling of product she made
- Sell her fabric (finished)

Jobs to be done

- When I sell my fabric I want to let the designer know that my products are recycled, circular and made with sustainable practices
- When I order yarn I want to know the composition so I can guarantee the quality
- When I get more orders that I can combine I can have more impact so I can push the industry towards something better
- When I go to a seller, I want to offer broadly possible products so I can show analytical reports supporting me.
- When someone buys my clothes, I want her to have access to all the story of the material, so I can improve the design value.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Amanda	**Age:** 28
Profession	Fashion Designer	
Level of IT-skills	Low	
Organisation name	Start-up / small brand	
Number of employees	2	**Location:** Amsterdam Center
Type of Organisation	Company ☒ Authority □ University □ Fablab □ Other □	

Actual challenges & problems

- She wants sustainable textiles and she does not know where to buy them
- She looks to grow the network (manufacture, Fablab, ...)
- (Bridge between design and technical language)

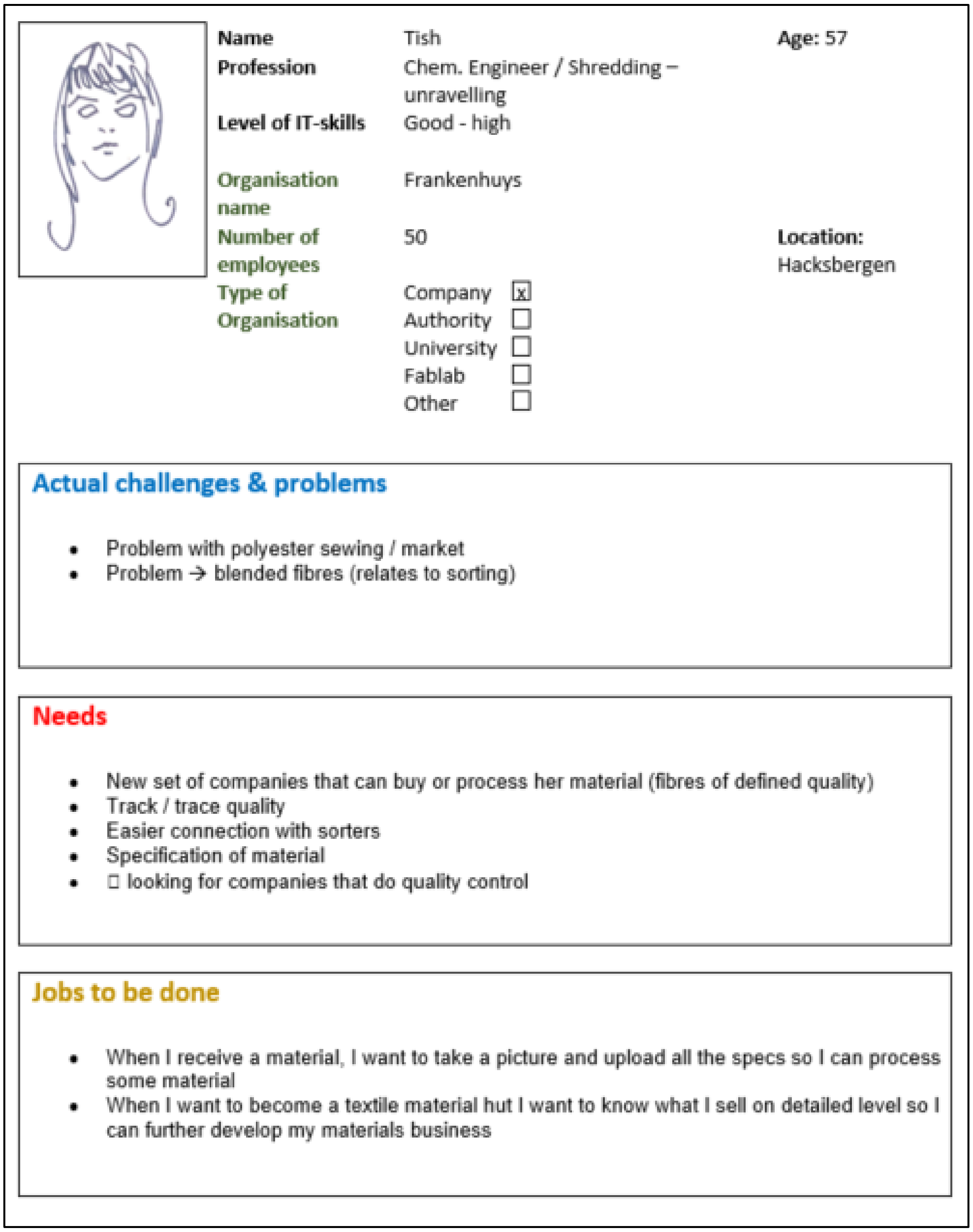
Needs

- Track material for value/ storytelling of what she makes
- She wants finished textile (no yearn/ no fibres) or the contact of a person who can provide them
- (she needs to know how to deal with competition in the field (differentiate))
- Specification about what & where
- Tags → connection to glossary (technical specifications etc.)

Jobs to be done

- When I communicate about my products I want to know the history of the material so I can be honest about my products
- When I want to connect with manufacturers I want to know where to look for them so I can expand my network
- When I sell a garment I want to let the consumer know where it came from
- When I buy sustainable textile I want to know where it comes from so I can let my buyer know and add value to my piece
- When I am looking for a good material I also want to get to know more / learn about the technical aspect of materials

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



	Name	Tish	**Age:** 57
	Profession	Chem. Engineer / Shredding – unravelling	
	Level of IT-skills	Good - high	
	Organisation name	Frankenhuys	
	Number of employees	50	**Location:** Hacksbergen
	Type of Organisation	Company ☒ Authority ☐ University ☐ Fablab ☐ Other ☐	

Actual challenges & problems

- Problem with polyester sewing / market
- Problem → blended fibres (relates to sorting)

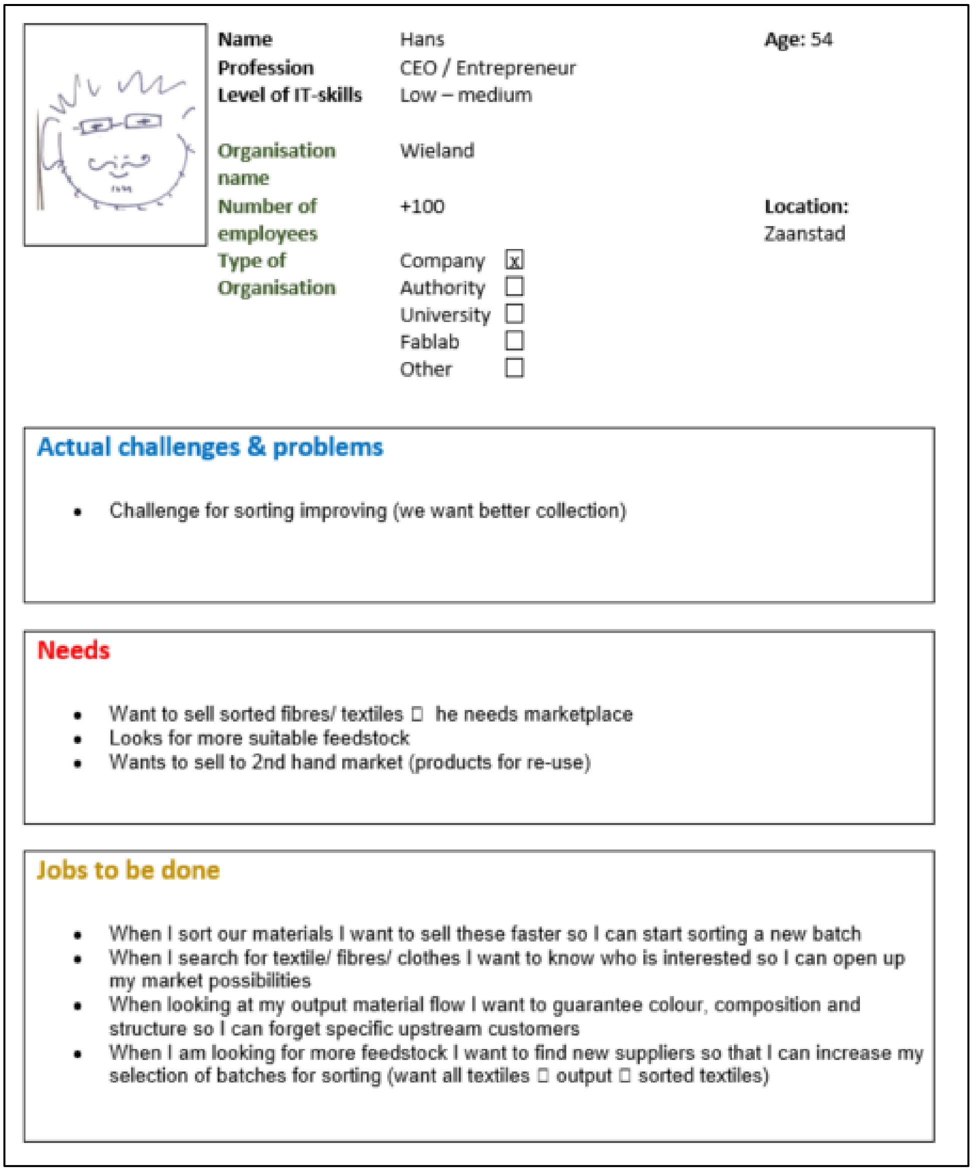
Needs

- New set of companies that can buy or process her material (fibres of defined quality)
- Track / trace quality
- Easier connection with sorters
- Specification of material
- □ looking for companies that do quality control

Jobs to be done

- When I receive a material, I want to take a picture and upload all the specs so I can process some material
- When I want to become a textile material hut I want to know what I sell on detailed level so I can further develop my materials business

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Hans		**Age:** 54
Profession	CEO / Entrepreneur		
Level of IT-skills	Low – medium		
Organisation name	Wieland		
Number of employees	+100		**Location:** Zaanstad
Type of Organisation	Company Authority University Fablab Other	☒ □ □ □ □	

Actual challenges & problems

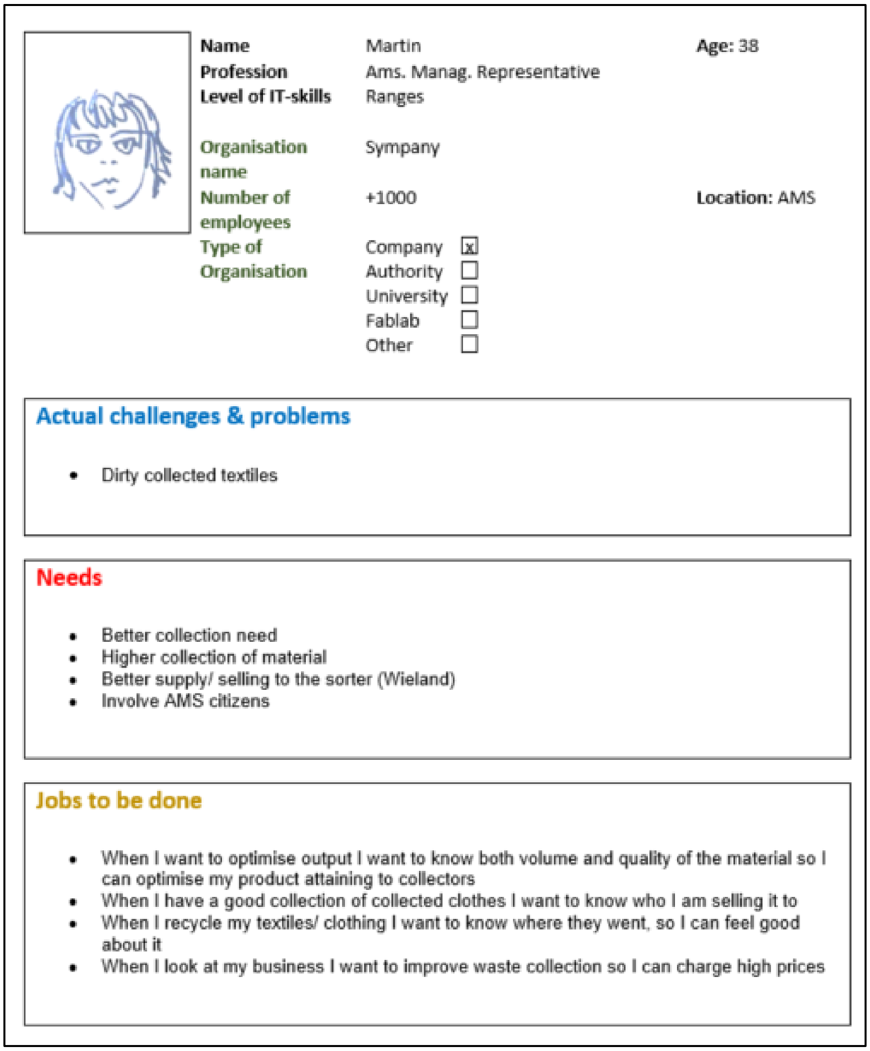
- Challenge for sorting improving (we want better collection)

Needs

- Want to sell sorted fibres/ textiles he needs marketplace
- Looks for more suitable feedstock
- Wants to sell to 2nd hand market (products for re-use)

Jobs to be done

- When I sort our materials I want to sell these faster so I can start sorting a new batch
- When I search for textile/ fibres/ clothes I want to know who is interested so I can open up my market possibilities
- When looking at my output material flow I want to guarantee colour, composition and structure so I can forget specific upstream customers
- When I am looking for more feedstock I want to find new suppliers so that I can increase my selection of batches for sorting (want all textiles output sorted textiles)



Name	Martin	**Age:** 38
Profession	Ams. Manag. Representative	
Level of IT-skills	Ranges	
Organisation name	Sympany	
Number of employees	+1000	**Location:** AMS
Type of Organisation	Company ☒ Authority □ University □ Fablab □ Other □	

Actual challenges & problems

- Dirty collected textiles

Needs

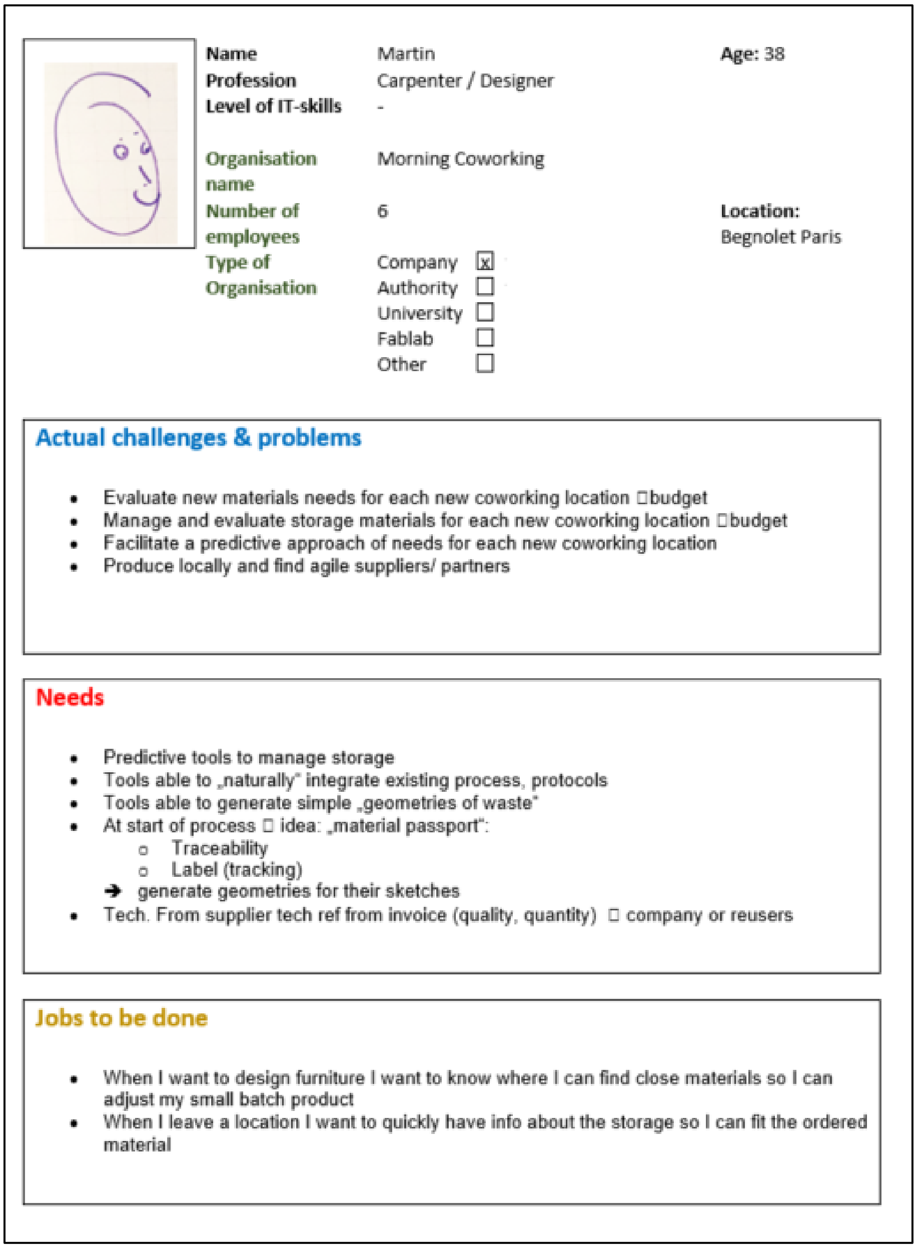
- Better collection need
- Higher collection of material
- Better supply/ selling to the sorter (Wieland)
- Involve AMS citizens

Jobs to be done

- When I want to optimise output I want to know both volume and quality of the material so I can optimise my product attaining to collectors
- When I have a good collection of collected clothes I want to know who I am selling it to
- When I recycle my textiles/ clothing I want to know where they went, so I can feel good about it
- When I look at my business I want to improve waste collection so I can charge high prices

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

6.3.2 Paris



Name	Martin	**Age:** 38
Profession	Carpenter / Designer	
Level of IT-skills	-	
Organisation name	Morning Coworking	
Number of employees	6	**Location:** Begnolet Paris
Type of Organisation	Company ☒ Authority ☐ University ☐ Fablab ☐ Other ☐	

Actual challenges & problems

- Evaluate new materials needs for each new coworking location □budget
- Manage and evaluate storage materials for each new coworking location □budget
- Facilitate a predictive approach of needs for each new coworking location
- Produce locally and find agile suppliers/ partners

Needs

- Predictive tools to manage storage
- Tools able to „naturally" integrate existing process, protocols
- Tools able to generate simple „geometries of waste"
- At start of process □ idea: „material passport":
  - Traceability
  - Label (tracking)

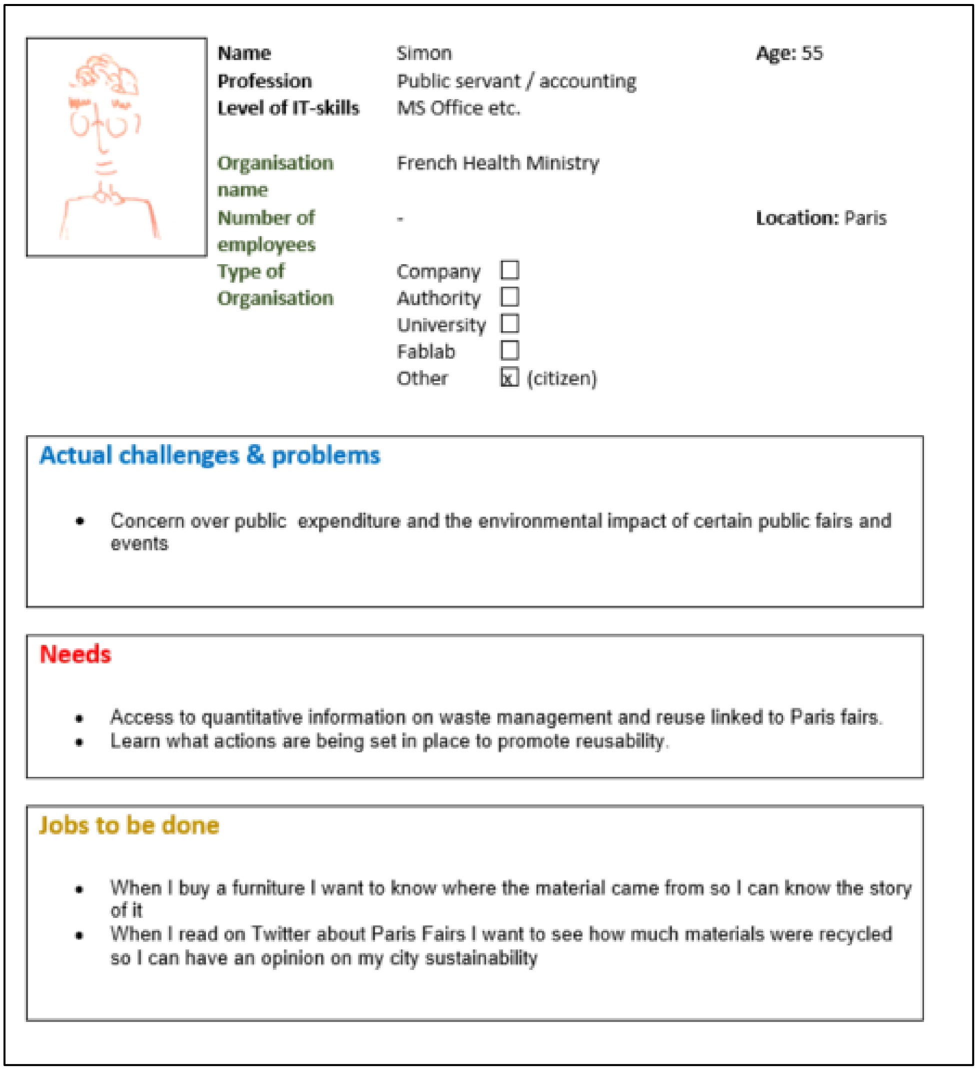
➔ generate geometries for their sketches

- Tech. From supplier tech ref from invoice (quality, quantity) □ company or reusers

Jobs to be done

- When I want to design furniture I want to know where I can find close materials so I can adjust my small batch product
- When I leave a location I want to quickly have info about the storage so I can fit the ordered material

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Simon	**Age:** 55
Profession	Public servant / accounting	
Level of IT-skills	MS Office etc.	
Organisation name	French Health Ministry	
Number of employees	-	**Location:** Paris
Type of Organisation	Company □ Authority □ University □ Fablab □ Other ☒ (citizen)	

Actual challenges & problems

- Concern over public expenditure and the environmental impact of certain public fairs and events

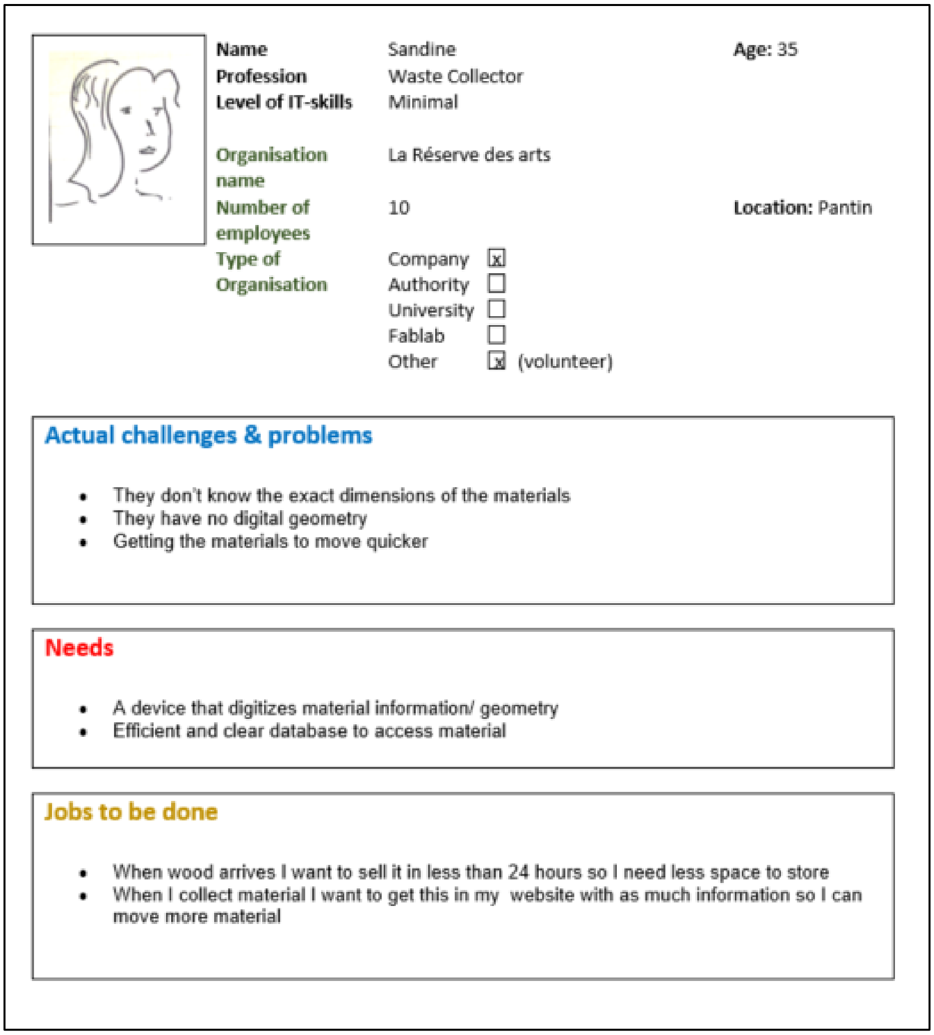
Needs

- Access to quantitative information on waste management and reuse linked to Paris fairs.
- Learn what actions are being set in place to promote reusability.

Jobs to be done

- When I buy a furniture I want to know where the material came from so I can know the story of it
- When I read on Twitter about Paris Fairs I want to see how much materials were recycled so I can have an opinion on my city sustainability

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Sandine	**Age:** 35
Profession	Waste Collector	
Level of IT-skills	Minimal	
Organisation name	La Réserve des arts	
Number of employees	10	**Location:** Pantin
Type of Organisation	Company ☒ Authority ☐ University ☐ Fablab ☐ Other ☒ (volunteer)	

Actual challenges & problems

- They don't know the exact dimensions of the materials
- They have no digital geometry
- Getting the materials to move quicker

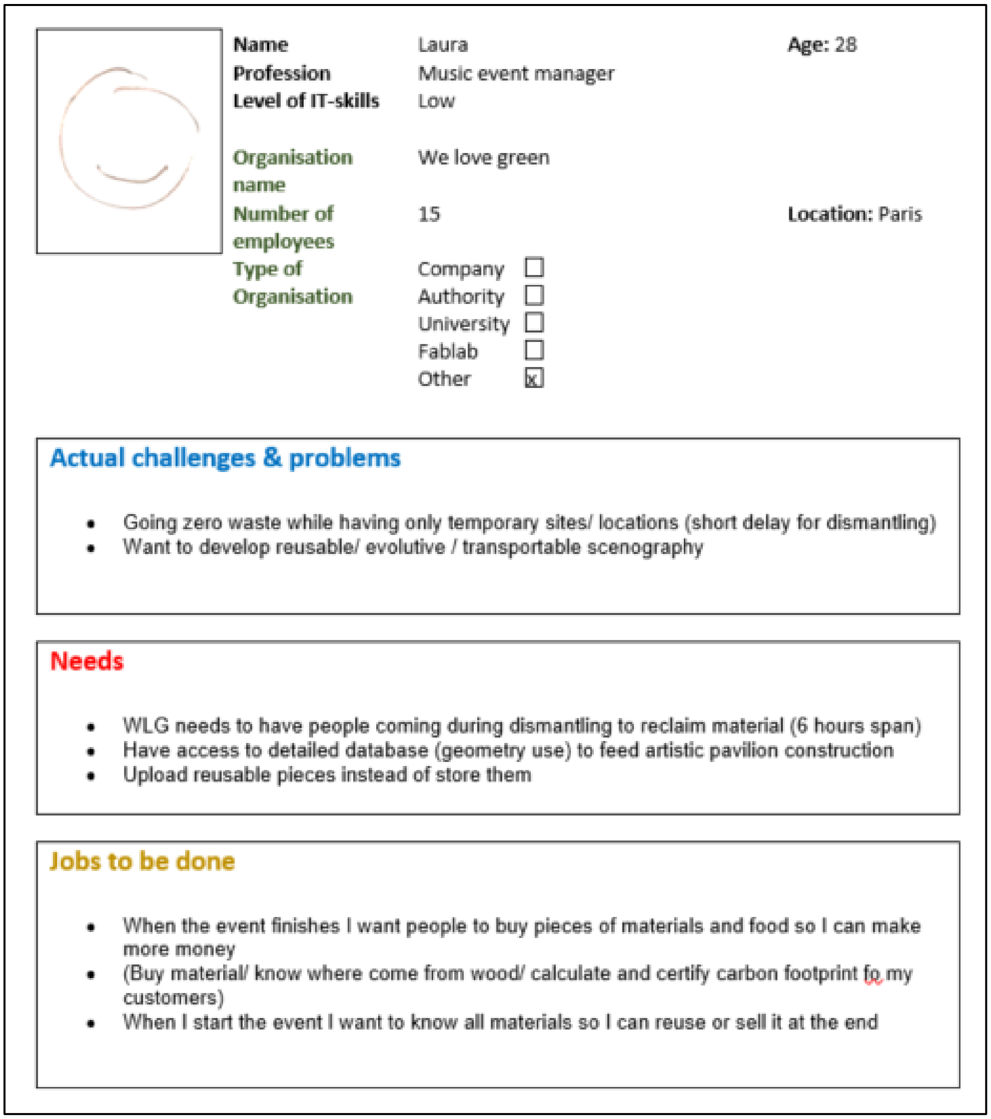
Needs

- A device that digitizes material information/ geometry
- Efficient and clear database to access material

Jobs to be done

- When wood arrives I want to sell it in less than 24 hours so I need less space to store
- When I collect material I want to get this in my website with as much information so I can move more material

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Laura	**Age:** 28
Profession	Music event manager	
Level of IT-skills	Low	
Organisation name	We love green	
Number of employees	15	**Location:** Paris
Type of Organisation	Company ☐ Authority ☐ University ☐ Fablab ☐ Other ☒	

Actual challenges & problems

- Going zero waste while having only temporary sites/ locations (short delay for dismantling)
- Want to develop reusable/ evolutive / transportable scenography

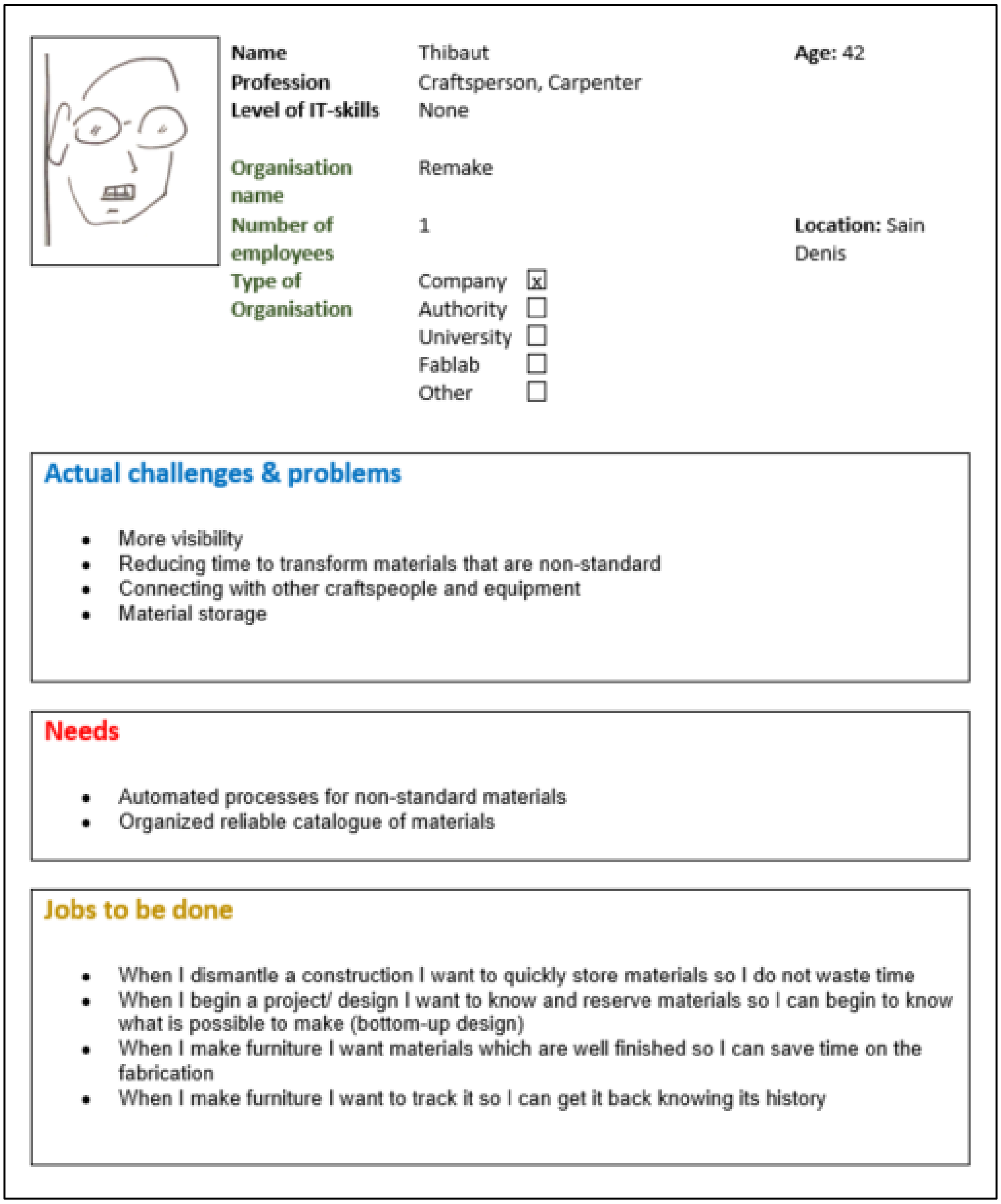
Needs

- WLG needs to have people coming during dismantling to reclaim material (6 hours span)
- Have access to detailed database (geometry use) to feed artistic pavilion construction
- Upload reusable pieces instead of store them

Jobs to be done

- When the event finishes I want people to buy pieces of materials and food so I can make more money
- (Buy material/ know where come from wood/ calculate and certify carbon footprint fo my customers)
- When I start the event I want to know all materials so I can reuse or sell it at the end

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Thibaut		**Age:** 42
Profession	Craftsperson, Carpenter		
Level of IT-skills	None		
Organisation name	Remake		
Number of employees	1		**Location:** Sain Denis
Type of Organisation	Company	☒	
	Authority	□	
	University	□	
	Fablab	□	
	Other	□	

Actual challenges & problems

- More visibility
- Reducing time to transform materials that are non-standard
- Connecting with other craftspeople and equipment
- Material storage

Needs

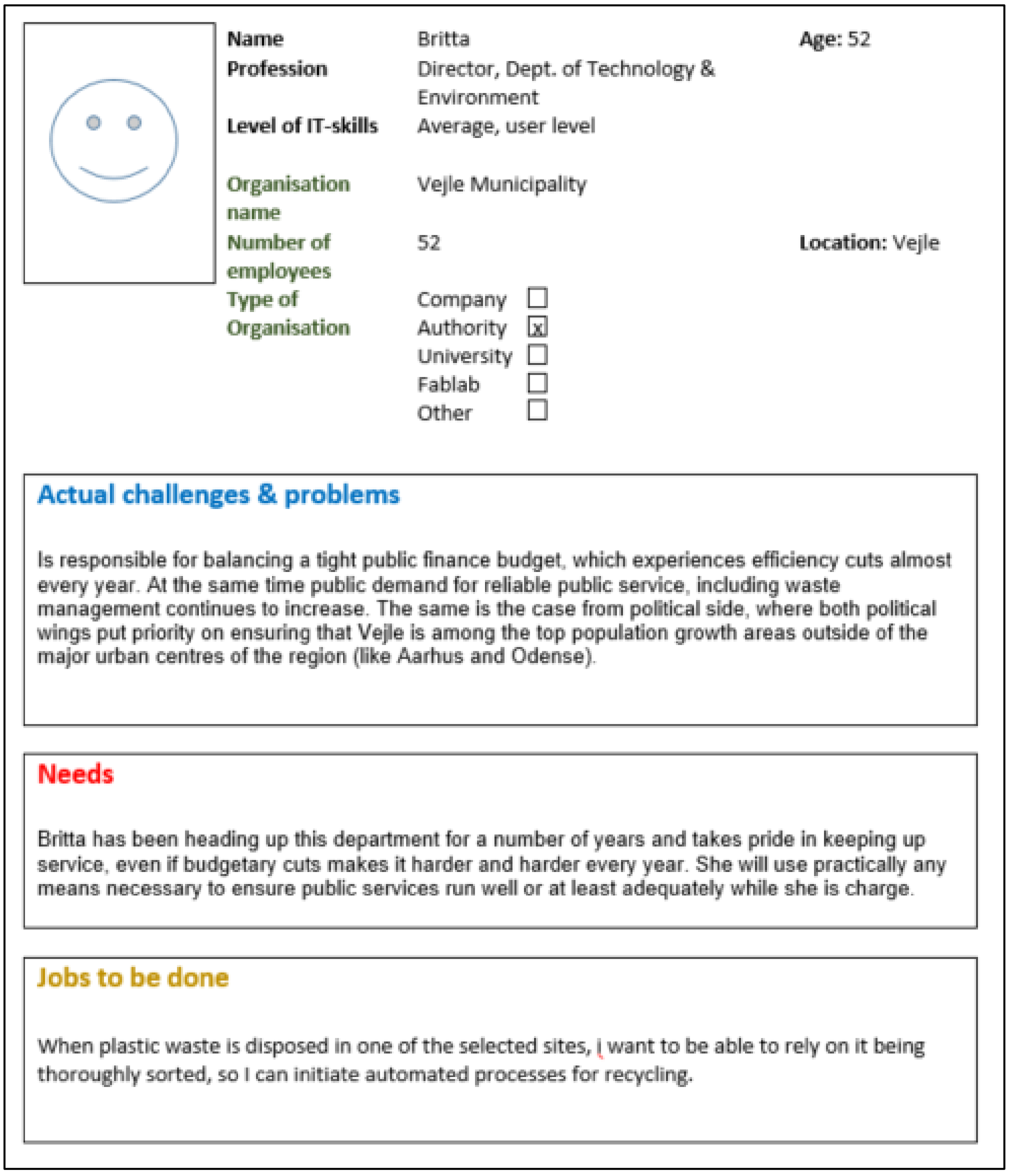
- Automated processes for non-standard materials
- Organized reliable catalogue of materials

Jobs to be done

- When I dismantle a construction I want to quickly store materials so I do not waste time
- When I begin a project/ design I want to know and reserve materials so I can begin to know what is possible to make (bottom-up design)
- When I make furniture I want materials which are well finished so I can save time on the fabrication
- When I make furniture I want to track it so I can get it back knowing its history

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

6.3.3 Vejle



Name	Britta	**Age:** 52
Profession	Director, Dept. of Technology & Environment	
Level of IT-skills	Average, user level	
Organisation name	Vejle Municipality	
Number of employees	52	**Location:** Vejle
Type of Organisation	Company ☐ Authority ☒ University ☐ Fablab ☐ Other ☐	

Actual challenges & problems

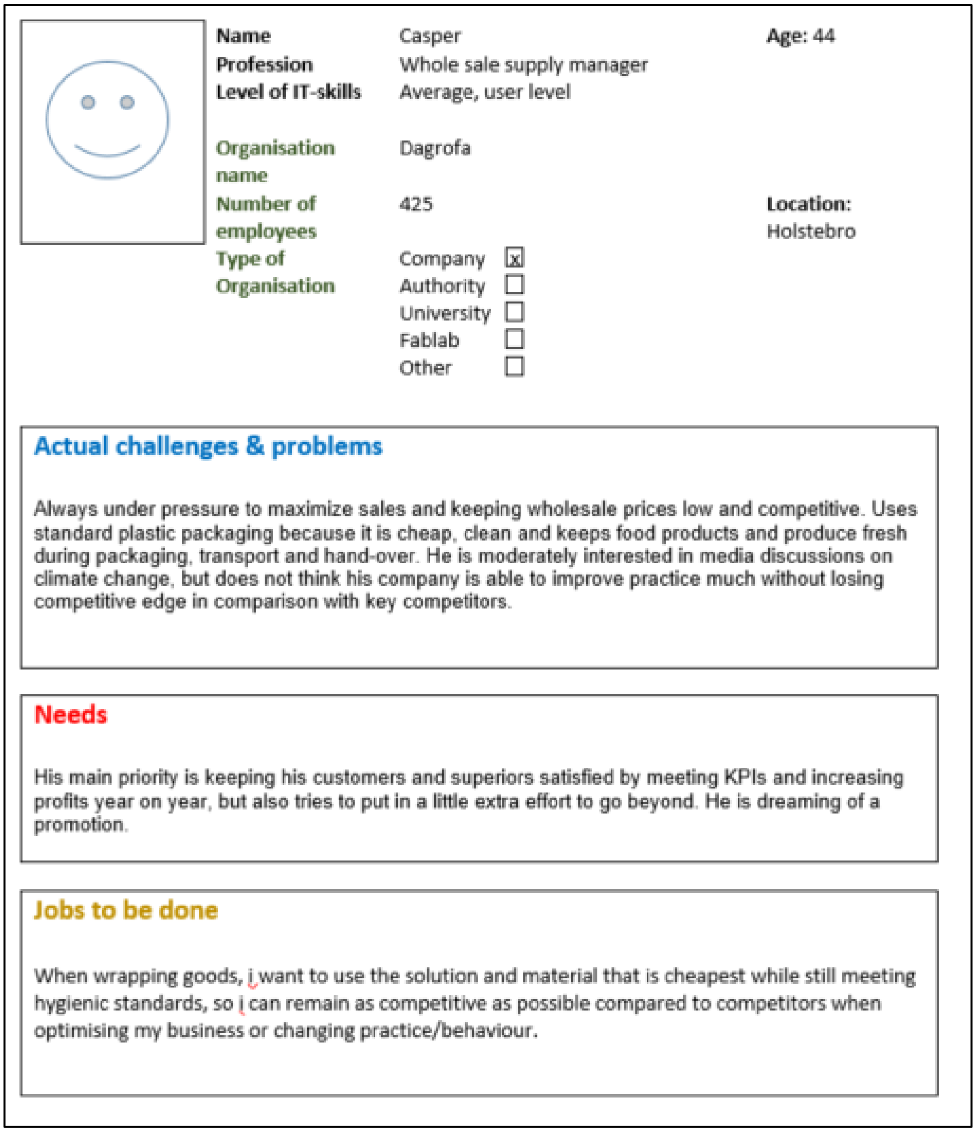
Is responsible for balancing a tight public finance budget, which experiences efficiency cuts almost every year. At the same time public demand for reliable public service, including waste management continues to increase. The same is the case from political side, where both political wings put priority on ensuring that Vejle is among the top population growth areas outside of the major urban centres of the region (like Aarhus and Odense).

Needs

Britta has been heading up this department for a number of years and takes pride in keeping up service, even if budgetary cuts makes it harder and harder every year. She will use practically any means necessary to ensure public services run well or at least adequately while she is charge.

Jobs to be done

When plastic waste is disposed in one of the selected sites, i want to be able to rely on it being thoroughly sorted, so I can initiate automated processes for recycling.



Name	Casper	**Age:** 44
Profession	Whole sale supply manager	
Level of IT-skills	Average, user level	
Organisation name	Dagrofa	
Number of employees	425	**Location:** Holstebro
Type of Organisation	Company ☒ Authority ☐ University ☐ Fablab ☐ Other ☐	

Actual challenges & problems

Always under pressure to maximize sales and keeping wholesale prices low and competitive. Uses standard plastic packaging because it is cheap, clean and keeps food products and produce fresh during packaging, transport and hand-over. He is moderately interested in media discussions on climate change, but does not think his company is able to improve practice much without losing competitive edge in comparison with key competitors.

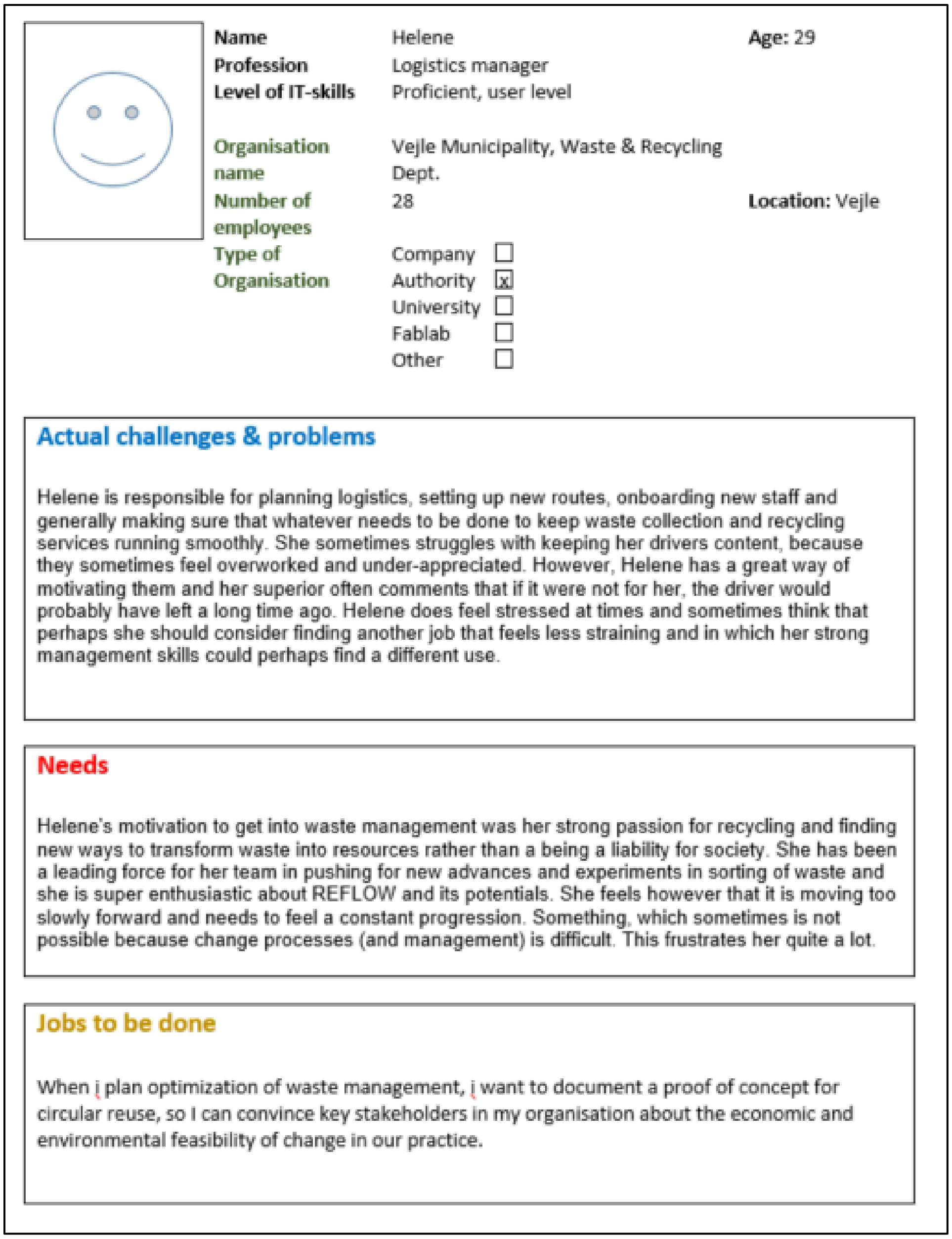
Needs

His main priority is keeping his customers and superiors satisfied by meeting KPIs and increasing profits year on year, but also tries to put in a little extra effort to go beyond. He is dreaming of a promotion.

Jobs to be done

When wrapping goods, i want to use the solution and material that is cheapest while still meeting hygienic standards, so i can remain as competitive as possible compared to competitors when optimising my business or changing practice/behaviour.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Helene	**Age:** 29
Profession	Logistics manager	
Level of IT-skills	Proficient, user level	
Organisation name	Vejle Municipality, Waste & Recycling Dept.	
Number of employees	28	**Location:** Vejle
Type of Organisation	Company ☐ Authority ☒ University ☐ Fablab ☐ Other ☐	

Actual challenges & problems

Helene is responsible for planning logistics, setting up new routes, onboarding new staff and generally making sure that whatever needs to be done to keep waste collection and recycling services running smoothly. She sometimes struggles with keeping her drivers content, because they sometimes feel overworked and under-appreciated. However, Helene has a great way of motivating them and her superior often comments that if it were not for her, the driver would probably have left a long time ago. Helene does feel stressed at times and sometimes think that perhaps she should consider finding another job that feels less straining and in which her strong management skills could perhaps find a different use.

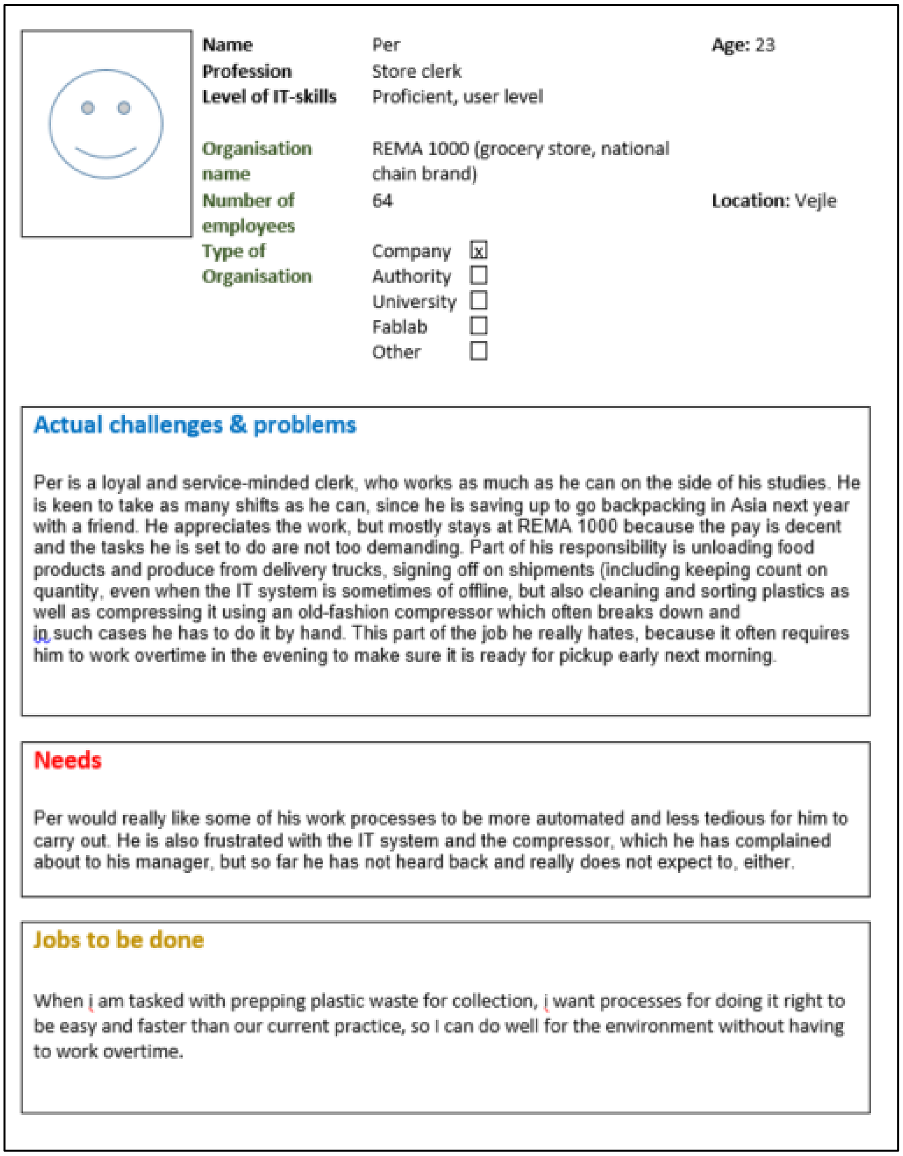
Needs

Helene's motivation to get into waste management was her strong passion for recycling and finding new ways to transform waste into resources rather than a being a liability for society. She has been a leading force for her team in pushing for new advances and experiments in sorting of waste and she is super enthusiastic about REFLOW and its potentials. She feels however that it is moving too slowly forward and needs to feel a constant progression. Something, which sometimes is not possible because change processes (and management) is difficult. This frustrates her quite a lot.

Jobs to be done

When i plan optimization of waste management, i want to document a proof of concept for circular reuse, so I can convince key stakeholders in my organisation about the economic and environmental feasibility of change in our practice.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Per	**Age:** 23
Profession	Store clerk	
Level of IT-skills	Proficient, user level	
Organisation name	REMA 1000 (grocery store, national chain brand)	
Number of employees	64	**Location:** Vejle
Type of Organisation	Company ☒ Authority □ University □ Fablab □ Other □	

Actual challenges & problems

Per is a loyal and service-minded clerk, who works as much as he can on the side of his studies. He is keen to take as many shifts as he can, since he is saving up to go backpacking in Asia next year with a friend. He appreciates the work, but mostly stays at REMA 1000 because the pay is decent and the tasks he is set to do are not too demanding. Part of his responsibility is unloading food products and produce from delivery trucks, signing off on shipments (including keeping count on quantity, even when the IT system is sometimes of offline, but also cleaning and sorting plastics as well as compressing it using an old-fashion compressor which often breaks down and in such cases he has to do it by hand. This part of the job he really hates, because it often requires him to work overtime in the evening to make sure it is ready for pickup early next morning.

Needs

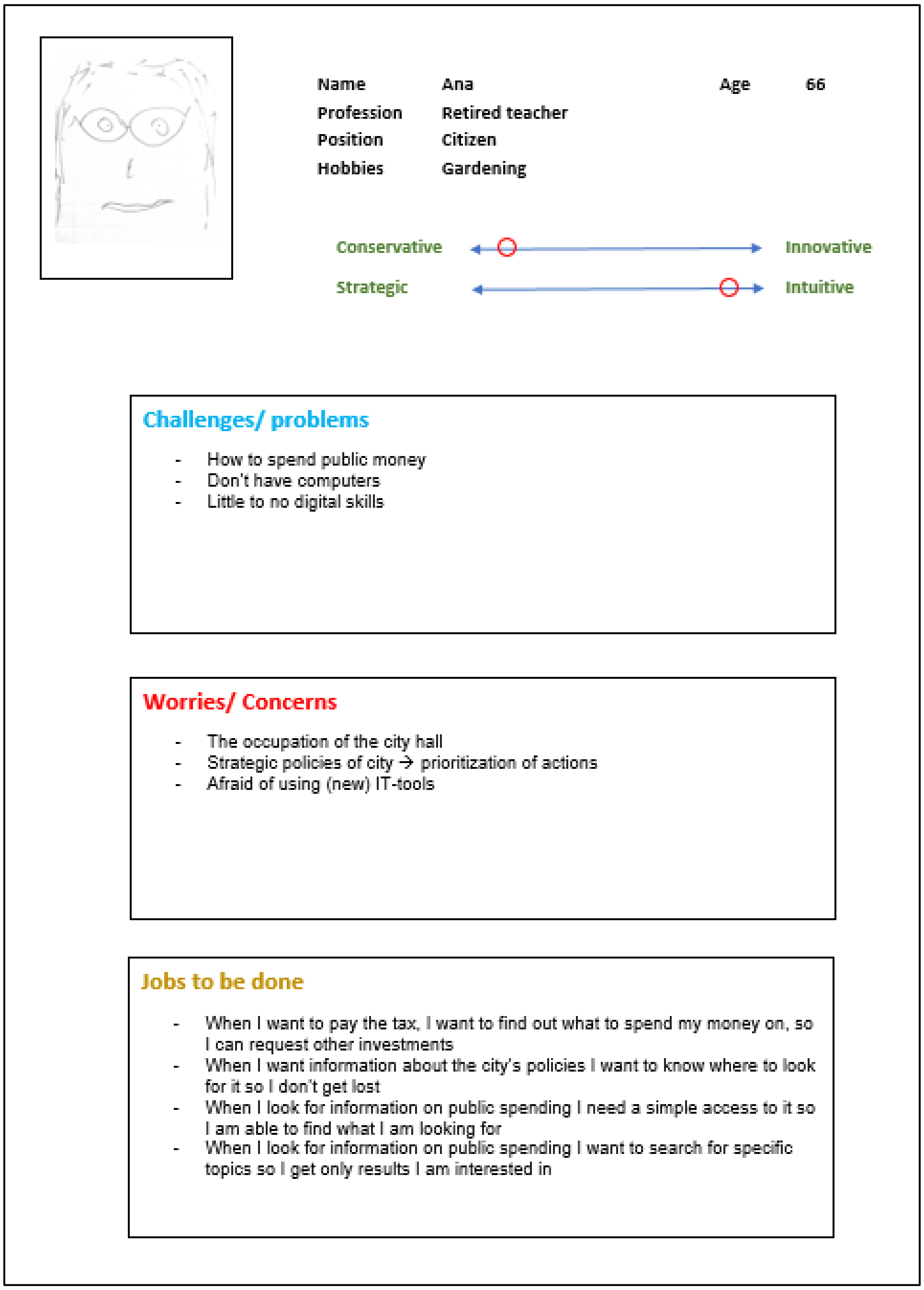
Per would really like some of his work processes to be more automated and less tedious for him to carry out. He is also frustrated with the IT system and the compressor, which he has complained about to his manager, but so far he has not heard back and really does not expect to, either.

Jobs to be done

When i am tasked with prepping plastic waste for collection, i want processes for doing it right to be easy and faster than our current practice, so I can do well for the environment without having to work overtime.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

6.3.4 Cluj



Name	Ana	Age	66
Profession	Retired teacher		
Position	Citizen		
Hobbies	Gardening		

Conservative ←○———————→ Innovative

Strategic ←———————○→ Intuitive

Challenges/ problems

- How to spend public money
- Don't have computers
- Little to no digital skills

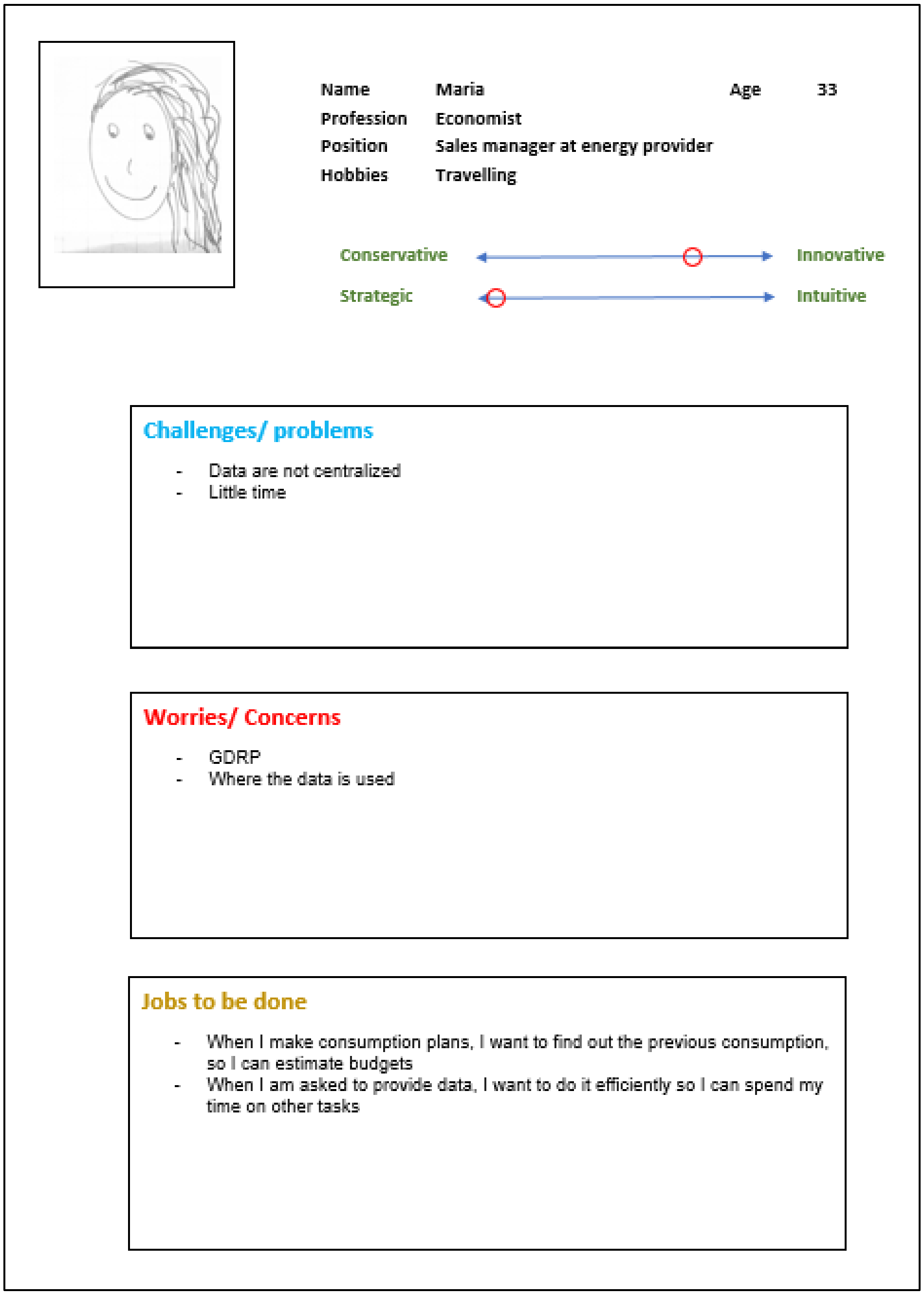
Worries/ Concerns

- The occupation of the city hall
- Strategic policies of city → prioritization of actions
- Afraid of using (new) IT-tools

Jobs to be done

- When I want to pay the tax, I want to find out what to spend my money on, so I can request other investments
- When I want information about the city's policies I want to know where to look for it so I don't get lost
- When I look for information on public spending I need a simple access to it so I am able to find what I am looking for
- When I look for information on public spending I want to search for specific topics so I get only results I am interested in

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Maria	Age	33
Profession	Economist		
Position	Sales manager at energy provider		
Hobbies	Travelling		

Conservative
Innovative
Strategic
Intuitive

Challenges/ problems

- Data are not centralized
- Little time

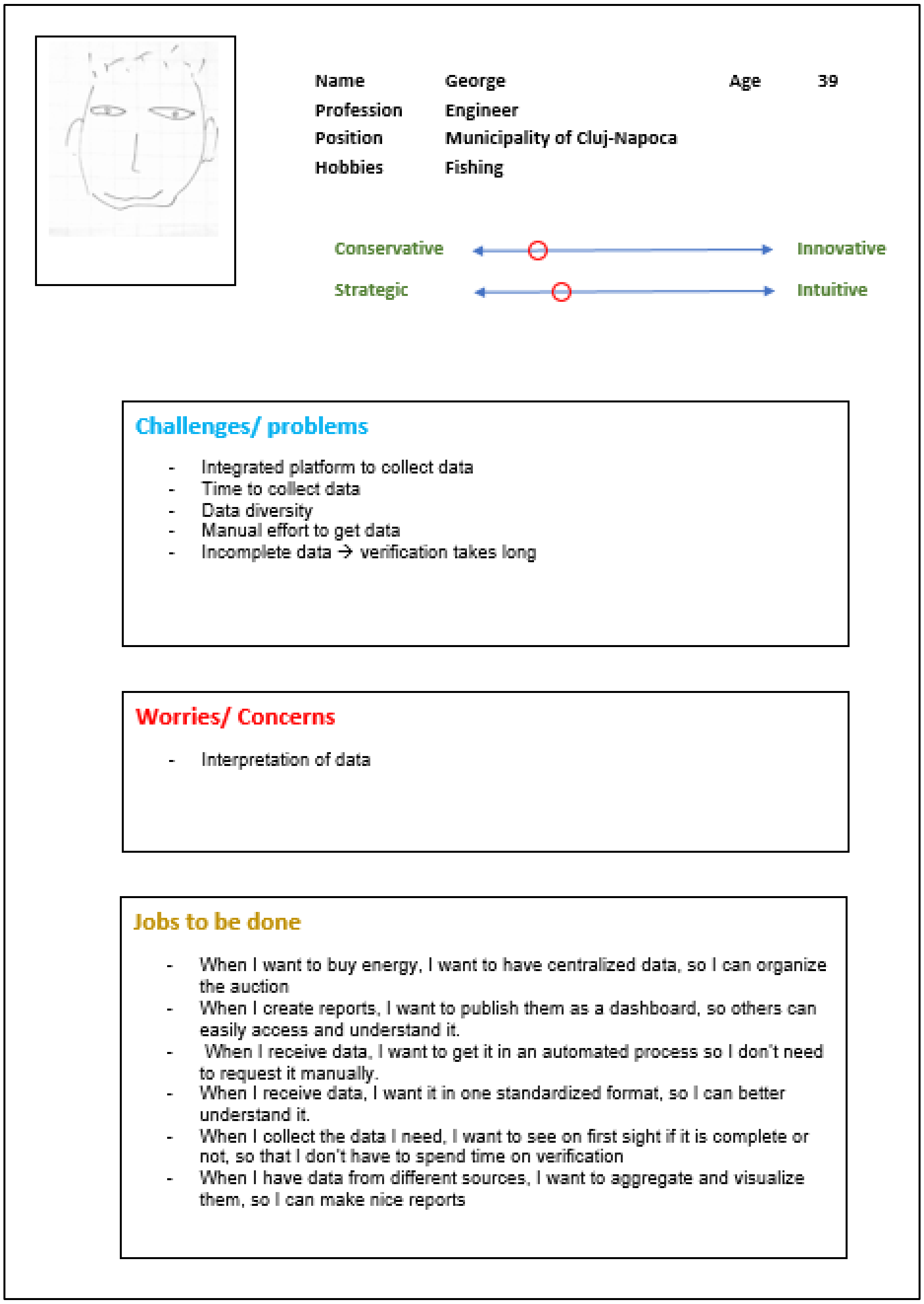
Worries/ Concerns

- GDRP
- Where the data is used

Jobs to be done

- When I make consumption plans, I want to find out the previous consumption, so I can estimate budgets
- When I am asked to provide data, I want to do it efficiently so I can spend my time on other tasks

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	George	Age	39
Profession	Engineer		
Position	Municipality of Cluj-Napoca		
Hobbies	Fishing		

Conservative ←○——————→ Innovative

Strategic ←—○—————→ Intuitive

Challenges/ problems

- Integrated platform to collect data
- Time to collect data
- Data diversity
- Manual effort to get data
- Incomplete data → verification takes long

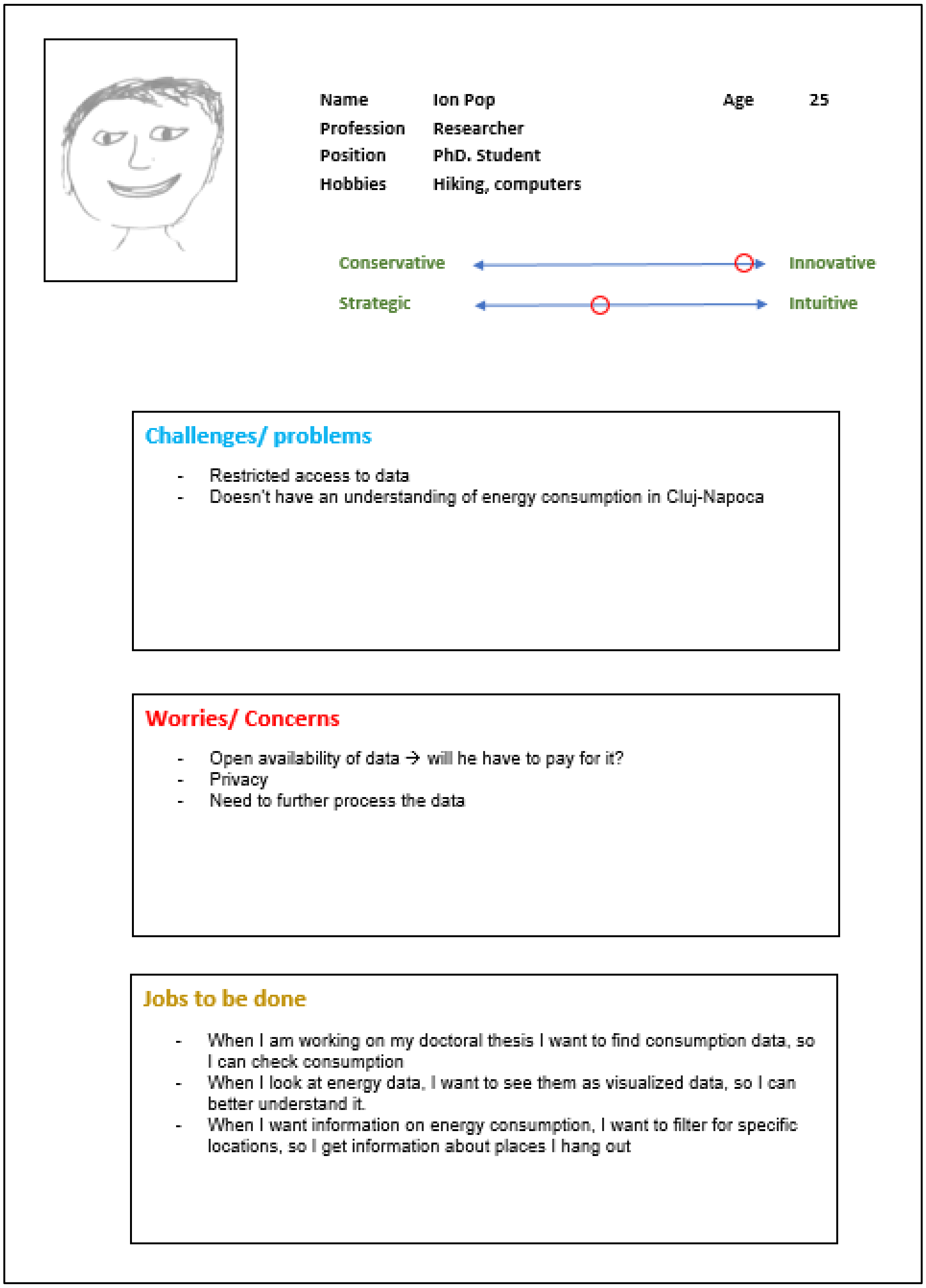
Worries/ Concerns

- Interpretation of data

Jobs to be done

- When I want to buy energy, I want to have centralized data, so I can organize the auction
- When I create reports, I want to publish them as a dashboard, so others can easily access and understand it.
- When I receive data, I want to get it in an automated process so I don't need to request it manually.
- When I receive data, I want it in one standardized format, so I can better understand it.
- When I collect the data I need, I want to see on first sight if it is complete or not, so that I don't have to spend time on verification
- When I have data from different sources, I want to aggregate and visualize them, so I can make nice reports

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.



Name	Ion Pop	Age	25
Profession	Researcher		
Position	PhD. Student		
Hobbies	Hiking, computers		

Conservative
Innovative
Strategic
Intuitive

Challenges/ problems

- Restricted access to data
- Doesn't have an understanding of energy consumption in Cluj-Napoca

Worries/ Concerns

- Open availability of data → will he have to pay for it?
- Privacy
- Need to further process the data

Jobs to be done

- When I am working on my doctoral thesis I want to find consumption data, so I can check consumption
- When I look at energy data, I want to see them as visualized data, so I can better understand it.
- When I want information on energy consumption, I want to filter for specific locations, so I get information about places I hang out

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

6.4 Use Cases

In this section the use cases for ReflowOS and the Open Data Dashboard are listed. The use cases are meant to be understood as a precision of the vision of both systems, showing how a user could possibly interact with the system, rather than prescribing the exact way of implementation. This is important to emphasize, because the use cases were created rather early during the project runtime and might have to be adjusted after some iteration phases.

6.4.1 Use Cases for ReflowOS

Name	**Login via web UI**	
ID	UC:ROS01	
Goal	Log in to ReflowOS to be able to use all functions.	
Brief Description	Starts when the user opens the login page, ends when the user is successfully logged in	
Actors	User with ReflowOS account	
Preconditions	The user is not logged in yet; The user has a ReflowOS account	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user visits the website of ReflowOS
	2	The user clicks on “login”
	3	The system displays a login mask
	4	The user inserts his username and password
	5	The system displays the inserted values (password is hidden and not displayed in plain text)
	6	The user clicks on “login”
	7	The system checks if username and password correspond
	8	After positive check: The user is successfully logged in.
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: Username or password incorrect	
	1	Same as basic flow steps 1-7
	2	The system notificates the user that the username or password are incorrect. The user is not logged in and asked to try again. An option for resetting the password is displayed.
Post-conditions	The user is logged in and has access to his profile and can perform all actions according to his / her assigned rights.	

Name	**Edit own profile information**
ID	UC:ROS02
Goal	Successful editing of own profile information

Brief Description	Starts when user has the desire to change his/her profile information. Ends when changes of profile information are successfully stored.	
Actors	Any user with ReflowOS account	
Preconditions	The user has to be logged in	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	User clicks on “Edit profile information”
	2	System displays a form with all stored profile information
	3	User clicks in a field he/she wants to edit (e.g. username, biography, profile picture)
	4	User enters new data into the field
	5	System checks whether data is valid
	6	After all changes have been done, user clicks on “save”
	7	System stores the changes and displays notification about successful change of profile information
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	Condition causing branching: User does not want to save the changes	
	1	Same as basic flow steps 1-5
	2	After entering the changes, the user decides to discards the changes and clicks on “cancel”
	3	System asks the user if he/she really wants to discard the changes
	4	User confirms
	5	System does not store the changes
	Condition causing branching: User enters invalid value (e.g. username already taken)	
	1	Same as basic flow steps 1-5
	5	System shows notification that the entered value is invalid
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: The changes cannot be stored by the system due to connection issues	
	1	Same as basic flow steps 1-6
	2	The system shows a notification that the changes could not be stored. The user is asked to try again later
Post-conditions	System stores the information changed	

Name	**Discover other users in the network**
ID	UC:ROS03
Goal	Find other users in the network

Brief Description	Starts when the user has the desire to find other users in the network, ends when the user successfully has found other users in the network	
Actors	All types of users with an account	
Preconditions	The user must be logged in in order to find other users.	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user clicks into the search field and performs a search with an empty string
	2	The system displays a list of all users signed up to the network
	3	The user clicks on one of the listed usernames
	4	The system displays the profile information of the selected user
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	User searches for a specific user	
	1	The user clicks into the search field and starts to enter a username
	2	The system displays the entered term
	3	The system shows a list of usernames that match the entered term
	4	Every time the user concretes the term (adds more letters), the system updates the list of matching usernames
	5	The user clicks on the username he / she was searching for
	6	The system displays the profile information of the selected username
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: no matching results	
	1	The user enters a username in the search field and performs the search
	2	The system has no matching username and displays a respective notification
Post-conditions	After performing the search the system has to display a list of matching results	

Name	**Follow other users in the network**
ID	UC:ROS04
Goal	To be able to keep track of actions carried out by another user in the network
Brief Description	Starts when the user has the desire to follow a specific user in the network and ends when the actions regarding this specific user are directly connected to the first user´s profile
Actors	All types of users with an account
Preconditions	The user must be logged in. The user should not already follow the other user
Basic flow	*Most common path of interactions between the user and the system*

	Step	*Action*
	1	The user clicks into the search field and starts to enter a username
	2	The system displays the entered term
	3	The system shows a list of usernames that match the entered term
	4	Every time the user concretes the term (adds more letters), the system updates the list of matching usernames
	5	The user clicks on the username he / she was searching for
	6	The system displays the profile information of the selected user
	7	The user clicks on 'follow'
	8	The system adds the selected user to the user's 'friends' menu
Alternate flow 1	The user finds the specific user on a visualisation module: List	
	Step	*Action*
	1	The user clicks on one of the visualisation modules (by pilot city)
	2	The system displays a list of the last actions performed in the system. Each action includes the initial resource, the action itself, the final resource and the user
	3	The user clicks on the selected user's name
	4	The system displays the profile information of the selected user
	5	The user clicks on 'follow'
Alternate flow 2	The user finds the specific user on a visualisation module: List	
	Step	*Action*
	1	The user clicks on one of the visualisation modules (by pilot city)
	2	The user clicks on 'map view'
	3	The system displays a map of the last actions performed in the system. Each action is shown as a pin on the map
	4	The user clicks on one action
	5	The system displays the specific action, including the initial resource, the action itself, the final resource and the user
	6-8	The same as in Alternate flow 3-5
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: no matching results	
	1	The user enters a username in the search field and performs the search
	2	The system has no matching username and displays a respective notification
Post-conditions	There is a 'friends' menu or section in the user´s profile where he/she can see the last actions performed by all followed users. After 'following' a user, the system will add the selected user to the user's 'friends' menu.	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Name	**Unfollow other users in the network**	
ID	UC:ROS05	
Goal	To stop keeping track of actions carried out by a specific user in the network	
Brief Description	Starts when the user has the desire to stop seeing the actions performed by a specific user displayed in his/her user profile and ends when the user´s profile no longer shows this information	
Actors	All types of users with an account	
Preconditions	The user must be logged in. The user must already follow at least another user of the network.	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user clicks on his/her profile page
	2	The system displays the user’s profile page
	3	The user clicks on his/her ‘friends’ menu
	4	The system displays a list with all the user’s friends (followed users)
	5	The user scrolls down and clicks on a name
	6	The system displays the profile information of the selected user
	7	The user clicks on ‘unfollow’
Alternate flow 1	*The user searches for a specific username*	
	Step	*Action*
	1-4	Same as the basic flow
	5	The user clicks on the search field and starts to enter a username
	6	Every time the user concretes the term (adds more letters), the system updates the list of matching usernames
	7	The user clicks on the username he / she was searching for
	8	The system displays the profile information of the selected user
	9	The user clicks on ‘unfollow’
Alternate flow 2	*The user finds the username on a visualisation module*	
	Step	*Action*
	1	The user clicks on one of the visualisation modules (by pilot city)
	2	The system displays a list of the last actions performed in the system. Each action includes the initial resource, the action itself, the final resource and the user
	3	The user clicks on the selected user’s name
	4	The system displays the profile information of the selected user
	5	The user clicks on ‘unfollow’
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: no matching results	
	1	The user enters a username in the search field and performs the search

	2	The system has no matching username and displays a respective notification
Post-conditions	There is a 'friends' menu or section in the user´s profile where he/she can see the last actions performed by all followed users. After unfollowing a selected user, the user´s 'friends' menu does not display the actions of the specific user anymore.	

Name	**Create custom taxonomies on resources**	
ID	UC:ROS06	
Goal	Adding own taxonomies to a resource	
Brief Description	Starts when the user wants to add further information about a resource in form of a taxonomy, ends when the taxonomy is stored by the system. In the basic flow, this use case describes how a taxonomy can be created on an offer. The process described applies analogously to other resources and actions.	
Actors	User that perform an action such as "offer" or "request"	
Preconditions	The user has to be logged in; the resources for which a taxonomy shall be created, is existing	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user opens the edit mode of the respective offer
	2	The systems displays the edit mode
	3	The user scrolls down to the field for taxonomies
	4	The user adds one or more taxonomies, e.g. "red", "cotton"
	5	The system displays the entered terms
	6	After all terms are entered, the user clicks on "save"
	7	The system stores the entered taxonomies and notificates the user about the successful storage
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	Condition causing branching: User wants to cancel the creation of taxonomies	
	1	Same as basic flow steps 1-4
	2	The user does not want to save the changes and clicks on "cancel"
	3	The system asks the user if he / she really wants to cancel
	4	The user confirms by clicking on "ok"
	5	The system deletes the entered terms, the entered terms are not saved
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: The taxonomies cannot be stored by the system due to connection issues	
	1	Same as basic flows steps 1-6

	2	The system shows a notification that the taxonomies could not be stored. The user is asked to try again later
Post-conditions	The taxonomies on the resource are stored by the system. Users that subscribed to these taxonomies will get notifications. The resource is listed in the results when someone searches for the respective taxonomy.	

Name	**Publish an offer of an available resource or skill**	
ID	UC:ROS07	
Goal	The successful publication of an offer for a resource or skill	
Brief Description	Starts when user wants to publish an offer, ends when the offer is published	
Actors	Seller	
Preconditions	The user is logged in	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	In the UI, the user clicks on the respective button to create an offer
	2	The system displays a form for the offer to be described. The displayed fields for metadata depend on the pilot specific UI
	3	The user fills in the information to describe the material
	4	The user uploads images of the resource offered
	5	The user clicks on “continue”
	6	The system displays a summary of the offer
	7	The user confirms the offer and clicks on “publish”
	8	The system stores the information and publishes the offer
	9	The system notifies the user that the offer has been successfully published
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	Condition causing branching: User wants to save the offer	
	1	Same as basic flow steps 1-3
	2	The user wants to interrupt the creation of the offer and clicks on “save”
	3	The system saves the offer and notifies the user about the successful saving
	4	When logged in, the user has access to all his/her saved offers and can continue
	Condition causing branching: User wants to cancel the creation of the offer	
	1	Same as basic flow steps 1-2
	2	The user does not want to create an offer anymore and closes the form by clicking on [x]

	3	The system asks the user if she / he really wants to cancel the creation of the offer
	4	The user confirms the cancellation
	5	The form is closed, the system does not save the offer
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: User missed a mandatory field in the description of the offer	
	1	Same as basic flow steps 1-5
	2	The system shows a notification that a mandatory field is missing. The field with the missing value is highlighted.
	3	The user adds the missing information
	4	Same as basic flow steps 5-9
	Condition for the exception: The offer cannot be stored by the system due to connection issues	
	1	Same as basic flow steps 1-7
	2	The system shows a notification that the offer could not be stored. The user is asked to try again later.
Post-conditions	The offer is stored and published by the system. The offer can be found by other users in the network.	

Name	**Publish a request for a resource or skill**	
ID	UC:ROS08	
Goal	The successful publication of a request for a resource or skill	
Brief Description	Starts when user wants to publish a request, ends when the request is published	
Actors	Purchaser	
Preconditions	The user is logged in	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	In the UI, the user clicks on the respective button to create a request
	2	The system displays a form for the request to be described. The displayed fields for metadata depend on the pilot specific UI
	3	The user fills in the information to describe what he / she is looking for
	4	The user clicks on “continue”
	5	The system displays a summary of the request
	6	The user confirms the request and clicks on “publish”
	7	The system stores the information and publishes the request
	8	The system notifies the user that the request has been successfully published
Alternate flows	*Alternate path through the system*	

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement number 820937.

	Step	*Action*
	Condition causing branching: User wants to save request	
	1	*Same as basic flow steps 1-3*
	2	The user wants to interrupt the creation of the request and clicks on "save"
	3	The system saves the request and notifies the user about the successful saving
	4	When logged in, the user has access to all his/her saved requests and can continue
	Condition causing branching: The user wants to cancel the creation of the request	
	1	Same as basic flow steps 1-2
	2	The user does not want to create the request anymore and closes the form by clicking on [x]
	3	The system asks the user if she / he really wants to cancel the creation of the request
	4	The user confirms the cancellation
	5	The form is closed, the system does not save the request
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: The user missed a mandatory field in the description of the request	
	1	Same as basic flow steps 1-4
	2	The system shows a notification that a mandatory field has been missed. The field with the missing value is highlighted
	3	The user adds the missing information
	4	Same as basic flow steps 5-8
	Condition causing exception: The request cannot be stored by the system due to connection issues	
	1	Same as basic flow steps 1-6
	2	The system shows a notification that the request could not be stored. The user is asked to try again later
Post-conditions	The request is stored and published by the system. The request can be found by other users in the network	

Name	**Start a thread on an existing offer or request**
ID	UC:ROS09
Goal	Posting a comment on an existing offer or request, e.g. for asking the owner questions. The thread is public and can be read by any user that is part of the network
Brief Description	Starts when a user wants to comment an offer or request, ends when the comment is posted

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

Actors	Any user that is logged in	
Preconditions	The user is logged in. There exists an offer or request to start a thread on	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user sees an offer or request that he or she would like to add a comment on
	2	The user clicks on “comment”
	3	The system displays a field for adding a comment
	4	The user enters the comment
	5	The system displays the entered values
	6	The user clicks on “send”
	7	The system saves the comment and publishes it. The comment is displayed as published
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	Condition causing branching: The user wants to cancel the start of a thread	
	1	Same as basic flow steps 1-5
	2	The user does not want to send the comment and clicks on “cancel”
	3	The system asks the user if he/she really wants to cancel the action
	4	The user confirms
	5	The system deletes the entered value and the field for adding a comment is closed
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: The comment cannot be stored by the system due to connection issues	
	1	*Same as basic flow steps 1-6*
	2	The system shows a notification that the comment could not be stored. The user is asked to try again later
Post-conditions	The comment /thread is stored by the system and displayed on the detailed view of the respective offer. Other users can see the thread and react to it by adding further comments	

Name	**Subscribe to one or more taxonomies**
ID	UC:ROS10
Goal	To be able to keep track of actions labelled with a certain taxonomy
Brief Description	Starts when the user has the desire to follow actions labelled with a certain taxonomy and ends when the subscription is successfully saved by the system and the user’s profile is connected to the taxonomies he / she subscribed to

Actors	All types of users with an account	
Preconditions	The user must be logged in. The user should not already be subscribed to the respective taxonomy	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user clicks into the search field and starts to enter a term (taxonomy)
	2	The system displays the entered term.
	3	The system shows a list of taxonomies that match the entered term
	4	Every time the user concretes the term (adds more letters), the system updates the list of matching taxonomies
	5	The user clicks on the taxonomy he/she was searching for
	6	The system displays the selected taxonomy and the option to subscribe to it
	7	The user clicks on “subscribe”
	8	The system adds the selected taxonomy to the user’s subscription list and notificates the user about the successful subscription
Alternate flows	The user finds a taxonomy in an action (e.g. offer or request)	
	Step	*Action*
	1	The user clicks on an offer or request (or another action which has taxonomies) created by another user
	2	The system displays the information about the offer or request, including the taxonomies the creator used to describe the resource
	3	The user clicks on one of the taxonomies
	4	The system displays the option to subscribe to the selected taxonomy
	5	The user clicks on “subscribe”
	6	The system adds the selected taxonomy to the user’s subscription list
Exception flows	*Exception handling by the system*	
	Step	*Action*
	No matching results	
	1	The user enters a taxonomy in the search field and performs the search
	2	The system has no matching taxonomy and displays a respective notification
	The subscription cannot be stored by the system due to connection issues	
	1	Same as basic flows steps 1-7
	2	The system shows a notification that the subscription could not be saved. The user is asked to try again later.
Post-conditions	IN the user’s profile, there is a subscription list where all taxonomies are listed which the user has subscribed to. After subscribing to a taxonomy, the system adds the respective taxonomy to the user’s list	

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement number 820937.

Name	**Search for material via web UI**	
ID	UC:ROS11	
Goal	Locate a resource in the system according to the user's needs	
Brief Description	It starts when the user is in need of a specific resource which may be in the system and it ends when the resource has been located.	
Actors	Purchaser	
Preconditions	User is logged in	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user clicks on one of the visualisation modules (by pilot city)
	2	The system displays a list of the last actions performed in the system. Each action includes the initial resource, the action itself, the action's date, the final resource and the user
	3	The user filters the list applying tags from a tag cloud (e.g. "offers", "wood", "red")
	4	The system displays a list with the actions filtered according to the user's criteria
	5	The user scrolls down and clicks on one specific action (a resource that is offered)
	6	The system displays the specific action, including the resource (material)
Alternate flow	The user searches for a specific resource (material)	
	Step	*Action*
	1	The user clicks into the search field and performs a search with an empty string
	2	The system shows a list of resources (materials) that match the entered term. Each resource includes the last action performed on it and the user who performed it
	3	Every time the user concretes the term (adds more letters), the system updates the list of matching resources (materials)
	4	The user filters the list applying tags from a tag cloud (e.g. "offers", "wood", "red")
	5	The system displays a list with the actions filtered according to the user's criteria
	6	The user scrolls down the listed resources (materials) and clicks on one of them
	7	The system displays the specific resource (material), including the list of actions performed on it and the users who performed them.
Exception flow	*Exception handling by the system*	
	Step	*Action*
	The resource is not found (alternate flow)	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

	1	The user enters a resource (material) term in the search field and performs the search
	2	The system has no matching resource (material) name and displays a respective notification
	There is not any resource after applying the tags	
	1	The user enters a combination of tags using the tag cloud when the resources are listed by the system
	2	The system has no resources (materials) listed for this combination of tags and displays a respective notification
Post-conditions	The user has located the resource and is able to see its characteristics	

Name	**Reserve material**	
ID	UC:ROS12	
Goal	Make a reservation of a certain amount of material	
Brief Description	Starts when the user has found material in ReflowOS which he/she would like to reserve. Ends when the reservation is confirmed by the system	
Actors	Purchaser	
Preconditions	The user has found material he/she wants to reserve. The user currently sees detailed information about the material. The material is still available and not reserved by another user	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	User clicks on “Reserve material”
	2	The system displays a dialog box asking how much of the material the user wants to reserve
	3	The user types in the amount of material he/she wants to reserve
	4	The user confirms the input by clicking on “ok”
	5	The system checks whether the requested amount of material for reservation is available
	6	The system asks the user for the duration of reservation by displaying a date picker
	7	The user selects a date from the picker
	8	The system copies the value and displays the date together with the amount of material in the dialog box
	9	The user confirms his/her request for reservation and clicks on “ok”
	10	The system confirms the submission of the request by showing a notification that the reservation is forwarded to the seller
Alternate flows	*Alternate path through the system*	
	Step	*Action*
	1	*Condition causing branching:* User is not logged in

	2	User clicks on “Reserve material”
	3	The system notifies the user that he/she needs to be logged in in order to make a reservation. A log-in form is displayed
	4	The user types in his/her credentials and confirms
	5...	Same as basic flow steps 2-9
Exception flows	*Exception handling by the system*	
	Step	*Action*
	1	Condition for the exception: The user wants to reserve a quantity of the material that is not available
	2	Same as basic flow step 1-4
	3	The system checks whether the requested amount of material for reservation is available
	4	The system notifies the user that the requested amount is not available
	5	The user can re-enter the amount or cancel the reservation process
Post-conditions	The information about the reservation has to be stored and forwarded to the seller of the material	

Name	**Trace resource flow via QR code**	
ID	UC:ROS13	
Goal	Get information about the flow of a respective resource and how it was processed in ReflowOS.	
Brief Description	Starts when a user scans a resource’s QR code, ends when the resources flow is displayed in ReflowOS.	
Actors	All users	
Preconditions	The flow of the resource must be described beforehand in ReflowOS via actions.	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	Action
	1	User scans QR code of respective resource
	2	User is forwarded to the website of ReflowOS
	3	Systems displays all information available about the resource (title, users, pictures, description)
	4	System lists the flow of the resource (shows all actions that have been undertaken in context of the respective resource)
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system*	
	Step	*Action*
	Condition for the exception: No information available	
	1	*Same as basic flow step 1 and 2*

	2	*The system displays a notification that there are no information available about the resource in ReflowOS.*
Post-conditions	-	

Name	**Transfer material to another user**	
ID	UC:ROS14	
Goal	To transfer a certain amount of material to another user	
Brief Description	It starts when a request for a certain amount of a resource is made by a buyer and it ends when the resource is shown in the buyer's profile	
Actors	Seller	
Preconditions	Seller is logged in. Request is published by buyer. Request is located by seller (ROS011).	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	Seller clicks on the request
	2	System displays the action (request) and all its characteristics (user who performs it, due date, resource).
	3	Seller clicks on "make an offer"
	4	System displays a box with a confirmation button and an empty string
	5	Seller writes a message to the buyer in the empty string and clicks confirmation button
	6	System displays a notification in the buyer's profile with the seller's message
	7	Buyer accepts the request
	8	System displays a notification in the seller's profile that says "offer accepted by nameofbuyer"
	9	System freezes the request (action) and labels it as "already accepted" in the visualisation module
	10	Seller clicks on his/her profile
	11	Seller selects his/her resource on his/her profile
	12	System displays the resource, including all its characteristics and two empty strings
	13	Seller enters amount he/she wants to transfer and username of buyer, and then clicks "transfer"
	14	System adds / increases the transferred amount in the buyer's profile
	15	System deletes / decreases the transferred amount in the seller's profile
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system*	
	Step	*Action*

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.

	The request is cancelled by the buyer	
	1-9	Same as basic flow
	10	The buyer logs in his/her profile page
	11	The buyer clicks on the request on his/her actions list
	12	The system displays the action (request) and all its characteristics
	13	The buyer clicks on “cancel request”
	14	The system displays a warning message
	15	The buyer confirms the cancellation
	16	The system sends a message to the seller, then removes the action from the seller’s profile, from the buyer’s profile and from the visualisation modules and lists
	The offer is not accepted by the buyer	
	1-6	Same as basic flow
	7	Buyer does not accept the offer and (optionally) enters message to seller in empty string
	8	System display buyer’s message in seller’s profile
	The seller doesn’t write a message to the buyer	
	1-4	Same as basic flow
	5	Seller clicks confirmation button
	6	System displays a warning asking to the user to confirm if she/he wants to send the offer without message
	7	Seller clicks “confirm”
	8-17	Same as basic flow 6-15
	The changes cannot be stored by the system due to connection issues	
	1-5	Same as basic flow
	6	The system shows a notification that the offer could not be stored. The user is asked to try again later.
Post-conditions	The resource is deleted / has decreased the agreed amount in the seller’s profile, and it is now shown in the buyer’s profile. The offer is tagged as "transferred” in the system. The transfer is added to the resource history. The transfer is added to the seller’s history. The transfer is added to the buyer’s history. After the transfer is implemented, the system will send a message to seller and buyer to rate each other.	

Name	**Keep track of the consumption of a certain resource**
ID	UC:ROS15
Goal	To keep the inventory updated by logging how much of a certain resource is consumed during a process

Brief Description	Starts when the user logs the consumption of a resource during an economic activity. Ends when the resource amount is updated.	
Actors	User	
Preconditions	The user is logged in, the resource is available, and a request on a specific resource was approved by seller (ROS07 and ROS015)	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user clicks on his/her profile
	2	The system has notified the user his/her request on a specific resource was approved by seller
	3	The user clicks on the resource name
	4	The system displays the characteristics of this resource, including all the actions performed on it
	5	The user logs the economic activities, selecting the economic resource
	6	The user inputs the amount
	7	The user adds a description
	8	The user confirms the action on the resource
	9	The system sends a message to the seller, notifying his/her resource has been consumed
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system*	
	Step	Action
	a. The user inputs an amount that is higher than the amount available for this specific resource	
	1	Same as steps 1-4 (basic flow)
	2	System displays a warning asking to the user to input the amount again
	b. There is no action performed on the resource after it was offered	
	1	The user clicks on his/her profile
	2	User selects the one specific resource from his/her resource list
	3	The system displays the characteristics of this resource, including all the actions performed on it before it was offered.
Post-conditions	The available amount of the resource is correctly updated	

6.4.2 Use Cases for Open Data Dashboard

Name	**Search for Datasets**
ID	UC:ODD01
Goal	User finds the dataset he / she is looking for via a search

Brief Description	Starts when the user browses on the search page. Ends when the user finishes the search.	
Actors	User	
Preconditions	none	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The platform shows a list of the latest modified datasets
	2	The user uses the search field in order to type in search terms
	3	The user submits the search
	4	The platform looks for metadata that has the search term in its title, description or tags
	5	The platform provides the found results to the user, sorted by the last modified date of the data
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system*	
	Step	Action
	a. The search cannot find any results	
	1	Same as steps 1-4 (basic flow)
	2	The Platform displays a message that informs the user that no results were found
Post-conditions	none	

Name	**View dataset details page**	
ID	UC:ODD02	
Goal	User wants to find out more about a dataset	
Brief Description	Starts when the user clicks on a dataset in the result page. Ends when the results page is displayed	
Actors	User	
Preconditions	Search results page shows at least one dataset result	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The system collects all metadata to this dataset and presents it on a dedicated page to the user. The shown metadata is specified in DCAT-AP [1] [https://joinup.ec.europa.eu/solution/dcat-application-profile-data-portals-europe/release/200]
Alternate flows	*Alternate path through the system: -*	

Exception flows	*Exception handling by the system*: -
Post-conditions	none

Name	**Upload dataset**	
ID	UC:ODD03	
Goal	User upload data and metadata. The uploaded data is searchable	
Brief Description	Starts when the user clicks the upload button. Ends when the user has uploaded the dataset.	
Actors	User	
Preconditions	The user is logged in, the user has upload permissions, the user is assigned to a catalogue	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The platform opens a wizard, that asks the user to fill in metadata that got specified by DCAT-AP (1) (1) https://joinup.ec.europa.eu/solution/dcat-application-profile-data-portals-europe/release/200
	2	The user fills in all metadata and submits the forms
	3	The last form asks for uploading or referencing data
	4	The user selects the data to be uploaded and submits the form
	5	The platform uploads the data and stores data and metadata in the system
Alternate flows	*Alternate path through the system:*	
	Step	*Action*
	The user wants to reference public accessible data	
	1	*Same as 1-3*
	2	*The user provides a link to public accessible data*
	3	The platform stores the metadata and the reference to the data resource in the system
Exception flows	*Exception handling by the system*: -	
Post-conditions	The created metadata is searchable by any other user	

Name	**Create visualisation**
ID	UC:ODD04
Goal	User wants to create a visualisation of a dataset in order to better understand it

Brief Description	Starts when the user clicks the create visualisation button on a dataset details page. Ends when the visualisation is created.	
Actors	User	
Preconditions	The user is logged in, the user has visualisation creation permissions, the user is assigned to the datasets catalogue	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The platform opens a wizard
	2	The wizard provide a set of standard charts the user can chose of
	3	The user defines which data from the data ressources shall be visualised
	4	The user defines what shall be displayed on which axis
	5	The user gives the visualisation a name and a description
	6	The user submits the visualisation
	7	The platform attaches the visualisation to the dataset
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system*: -	
Post-conditions	The visualisation is accessible via a public URL, it can also be used for dashboards in UC:ODD006	

Name	**Create dashboard**	
ID	UC:ODD05	
Goal	User wants to create a dashboard in order to get a better understand of various KPIs	
Brief Description	Starts when the user visits the search page. Ends when the dashboard is created.	
Actors	User	
Preconditions	The user is logged in, the user has dashboard creation permissions, at least one dataset with visualisation is available	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	*Action*
	1	The user searches for data he / she needs for the dashboard
	2	The user clicks on a found dataset
	3	The Details page opens
	4	The user selects the visualisation that is needed for the dashboard
	5	The dataset visulalisation is added to the a basket
	6	The user clicks on the create dashboard button in the basket
	7	The platform opens a dashboard creation wizard
	8	The user provides a name and a description for the dashboard

	9	The user creates the dashboard
	10	The platform provides a canvas where the user can place the selected visualisations
	11	The user places the visualisations per drag & drop
	12	The user saves the dashboard
	13	The platform stores the dashboard
Alternate flows	*Alternate path through the system: -*	
	Step	*Action*
	The user wants to combine data from multiple datasets	
	1	*Same as 1 - 5*
	2	*The user goes back to the search page*
	3	*The user selects another dataset*
	4	*Same as 3 – 13 from regular flow*
Exception flows	*Exception handling by the system*: -	
Post-conditions	The dashboard is accessible via a public URL and searchable in the portal	

Name	**Filter for Dataset**	
ID	UC:ODD006	
Goal	User gets a filtered list of results for datasets	
Brief Description	Starts when the user has the desire to refine the list of datasets, ends when the user gets a filtered list	
Actors	User	
Preconditions	User already sees a list of datasets, e.g. after submitting a search	
Basic flow	*Most common path of interactions between the user and the system*	
	Step	
	1	*The user selects one or more filters which are presented next to the list of datasets. Filters could apply to e.g. license, region or category*
	2	*When the user clicks on the filter he / she wants to use, the system displays a drop down list with different options the user could filter for*
	3	*The user selects at least one option he / she wants to filter for*
	4	*The system displays the selected option*
	5	*The user clicks on "apply"*
	6	*The system filters according to the selected option*
	7	*The systems displays the results according to the selected filter*
Alternate flows	*Alternate path through the system: -*	
Exception flows	*Exception handling by the system: -*	
Post-conditions	none	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 820937.