D5.2- Automotive part data exchange platform

Project acronym:	**ReCiPSS**
Project full title:	**Resource-efficient Circular Product-Service Systems — ReCiPSS**
Grant agreement no.:	**776577-2**
Author/s:	**Dana Oniga, Markus Wagner, Adrian Dragomir, Cristina Prunaru**
Reviewed:	**SIG, C-ECO & KTH**
Approved:	**Amir Rashid**
Document Reference:	**D5.2**
Dissemination Level:	**PU**
Version:	**2**
Date:	**30.11.2019 04.03.2020 (resubmission)**

This is a draft document and subject to approval for final version. Therefore the information contained herein may change.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

History of Changes

Version	*Date*	*Modification reason*	*Modified by*
0.1	01.09.2019	Initial Draft	Dana Oniga, Cristina Prunaru
0.2	22.10.2019	Business content and demo-background	Markus Wagner
0.3	28.10.2019	Business content	Adrian Dragomir
0.4	05.11.2019	Draft review before document circulation	C-ECO
0.5	08.11.2019	Final document review by C-ECO	C-ECO
1.0	04.12.2019	Final reviewed deliverable	SIV
2.0	04.03.2020	Final version for resubmission	SIV

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

Table of contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

List of figure

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

List of abbreviations

Abbreviation	*Explanation*
API	Application Programming Interface
CE	Circular Economy
CPU	Central Processing Unit
D5.2	Deliverable 5.2
EC	European Commission
ERP	Enterprise Resource Planning
ICT	Information and Communications Technology
IoT	Internet of Thing
IT	Information Technology
MS6	Milestone 6
OE	Original Equipment
OS	Operating System
TL	Trade Level
T5.1	Task 5.1
T5.2	Task 5.2
UI	User Interface
WP	Work package

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

1. Executive Summary

This deliverable represents the outcome of the development work performed until now within WP5- ICT Platform for Multiple Life Cycles, under T5.2 “Implementing a part data management platform including standard data exchange protocols, ensuring transparency and efficiency of trade networks and reverse logistics.

The main purpose of the deliverable is to present the results obtained so far in developing a new platform and the functionalities still to be done until milestone **MS6: ICT platforms ready** in month 24 (May 2020). In the first eleven months since task T5.2 began, the main functions of the automotive part data exchange platform were developed: User-Management, View Options, Create options and Transfer multiple options. The newly created automotive part data management platform was connected to CoremanNet system.

The work performed in WP5, T5.2, will serve as base for WP7: Automotive parts demonstrator, for the deployment of the platform and for further improvements.

This report will emphasize the demo scope and description, will answer questions regarding the background of developing the automotive platform and the vision of its future and will demonstrate the connection with the circular business models. This document will describe the newly developed functionalities of the automotive part data management platform and how it will look at the end of the project. While all IT-components supporting the physical-processes or customer-interaction (e.g. core-evaluation-system or customer-data and reports, material-master-data) are already existing in C-ECO’s CoremanNet-system, the newly developed ReCIPSS-platform focuses solely on all functions needed to incorporate and manage return-options. As a result of T5.2, a demonstration was performed during the General Assembly in BRNO (20-22 November 2019), emphasizing the main functionalities of the automotive platform.

The main chapters in the document are: demo scope and description, the overview of the architecture used and the development methodology, a brief presentation of the functionalities developed in the platform and a more detailed presentation of the functional components.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

2. Introduction

This section presents the scope of the demo and its description, as well as the developments for Automotive part data exchange platform and link to Circular Economy**.**

2.1. Demo Scope and Description

This deliverable- D5.2 Automotive part data exchange platform and the demo are the results of T5.2:-Implementing a part data management platform including standard data exchange protocols ensuring transparency and efficiency of trade networks and reverse logistics, supporting the management of cores in terms of transparency, usage rate and financial optimization through the wholesale network, and the direct return of cores to the remanufacturers from any trade level (bypassing other trade levels).

The Automotive part data exchange platform will improve an existing circular product–service system by introducing a centralized clearinghouse that will allocate the related value of the transaction to each trade level, instead of each trade level clearing a transaction when processing a used spare part. The new platform will provide open and standardized protocols to ensure transparency throughout the value chain, as opposed to the situation when different trade levels were using heterogeneous IT systems to access part-related information.

2.1.1. General Business Background

In today's automotive aftermarket, remanufactured parts are sold from the producer (remanufacturer) towards the user (car-driver) via various trade-levels to a workshop, which is then actually applying the reman-part to the car, as illustrated in **Error! Reference source not found.**. In doing so, a used part (referred to as "core") is dismantled from the car, being then available at the workshop and needs to be transported backwards to the remanufacturer for being used in reman-production.

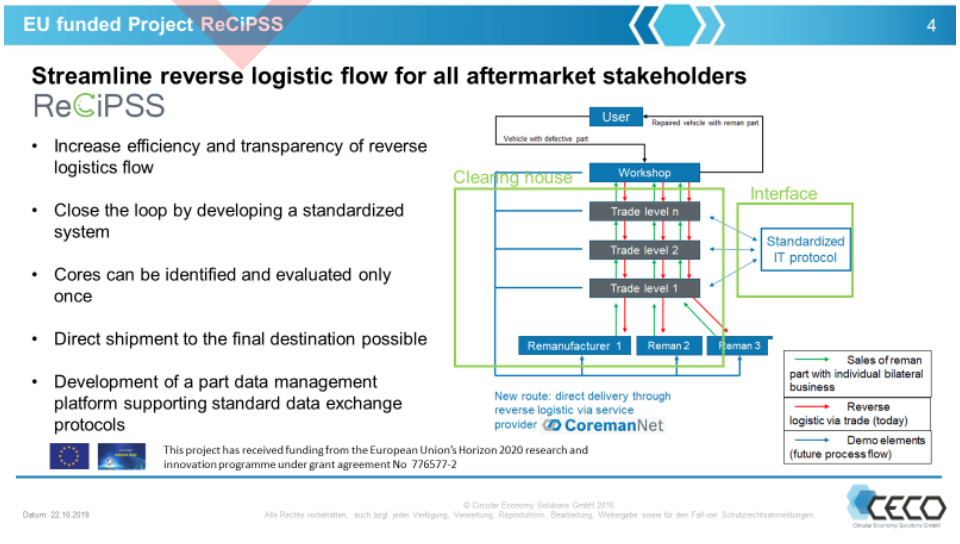


Figure 1: Streamline reverse logistic flow for all aftermarket stakeholders

To incentivize the return-flow of the cores, today a financial incentive in the sense of a deposit is commonly used in independent automotive aftermarket. This means that a buyer of a reman-part is paying the seller for the part itself plus an additional surcharge (deposit) which will be credited back in the event that the buyer is returning a core matching to reman-part they have purchased. “Matching” in this context means that the core needs to comply with a set of technical and commercial criteria in order to be acceptable as return and therefore also being credited. In today’s market the reman-producer is defining the return-criteria and the surcharge applied in the market, whereas the trade-levels in between usually just accepting and executing these rules with their customers. As the 1TL-traders are usually working with several reman-producers, it leaves the burden to them to handle the complexity of various rules and criteria for different reman-suppliers. Some traders which are specialized regionally or product-wise are purchasing from other traders. In that case, there might be several trade-levels involved until the reman-part is transferred from the producer to the workshop. And on every level, the technical and commercial return-rules plus the surcharge is applied. That means in the event of a core-return, every stakeholder in that chain needs to handle and check the parts and to decide whether to accept and credit their customers for the core-return. At the same time, they have to sort the cores and send them to the next destination from where they expect to be credited the surcharge. As common, commercially available ERP-systems are usually not supporting this core-reverse-flow, it is a complex problem for traders to determine where to actually send the used part and keep track on that. In some cases, especially at the last trade levels, ERP-systems are not existing at all. Here the book-keeping is conducted very manual, which is adding more complexity to the problem.

The overall system relies on a “right to return” associated with the core-surcharge as a fixed price. This “right to return” is in today’s market environment only existing and valid between the two business-partners which were initially trading a reman-part. The “right to return” is hardly fixed in a contract-agreement nor can it be transferred or executed by anybody else then those two original business-partners. It is implicitly accepted by the business-partners that the “right to return” a core is transferred with the reman-part.

However, this transfer of the “right to return” to a 3rd-party gets complex to track when several trade levels and OEMs are involved. Today there are no IT solutions available in the market that can manage the transfer of ‘right to return’ and associated ‘financial transaction’.

The above discussion highlights the major issues that are involved in the reverse logistics and management of cores. These issues are considered as major obstacles to implement circular economy principles in automotive sector. This automotive part data exchange platform is an essential development taken up by the ReCiPSS project.

By taking as weak points of the current situation the bilateral relationships that needed, the lack of IT infrastructure and the manual work which required to track all the “rights to return” among the stakeholders, within ReCiPSS an IT-platform will expand the idea of the “right to return” from the close bilateral relationship between two traders to the overall commercial ecosystem and will support on the tracking of all transactions. The way to describe the “right to return” for a core by the commercial instrument of a **financial option (warrant)**. This option should be

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

explicitly described and generated in the form of a digital entity which can be transferred to new owners independently from the physical item (reman-part or core) and being executed by them. The IT-development in ReCiPSS-automotive demonstrator is focusing to design and build the digital infrastructure to enable the handling of options and manage the core (material) flow more efficient and on time. In doing so, this will allow trade-users to gain more control over the reman- and core-business, allow optimization of logistical flows independently from the financial flows and enable the creation of a new market for return-options to be traded independently. The platform will also allow include relevant stakeholders such as independent core brokers to participate which are providing important benefit to remanufacturing-industry, but do not receive much digital support for their business as per today. Providing value to them as well in using the platform can increase and stabilize access to required cores making remanufacturing business. Furthermore, the platform will increase transparency among the stakeholders creating better business environment.

2.1.2. Commercial concept of options

An option is defining the right to buy or sell a defined good at a later time for a defined price. This means an option (or warrant) is a standardized contract which can be traded and transferred as a product as such.

An option is defined by the characteristics as shown below in **Error! Reference source not found.**.

First must be distinguished whether the option is granting the owner the right to sell ("put-option") or the right to buy ("call-option") something. Depending on the time-frame for execution related to the option, it can be distinguished as "European"-type or "American"-type. Whereas a European option can only be executed on its specific due-date (e.g. Option can be executed exactly on 31.12.2019. No sooner or later.) and an American option can be executed on any date prior to the due-date (e.g. Option can be executed anytime before 01.01.2020). Execution means the owner of the option can either buy (call) or sell (put) the so called underlying for a predefined price which is also part of the options-contract. The predefined price is referred to as the nominal value of the option. That is not to be mistaken with the market-price of the option in case it is traded. The nominal value is the price which has to be applied on the underlying in the event of the execution of the option, whereas the option-price or market-value is what stakeholders are willing to pay to acquire the right defined by the option (transfer of option).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

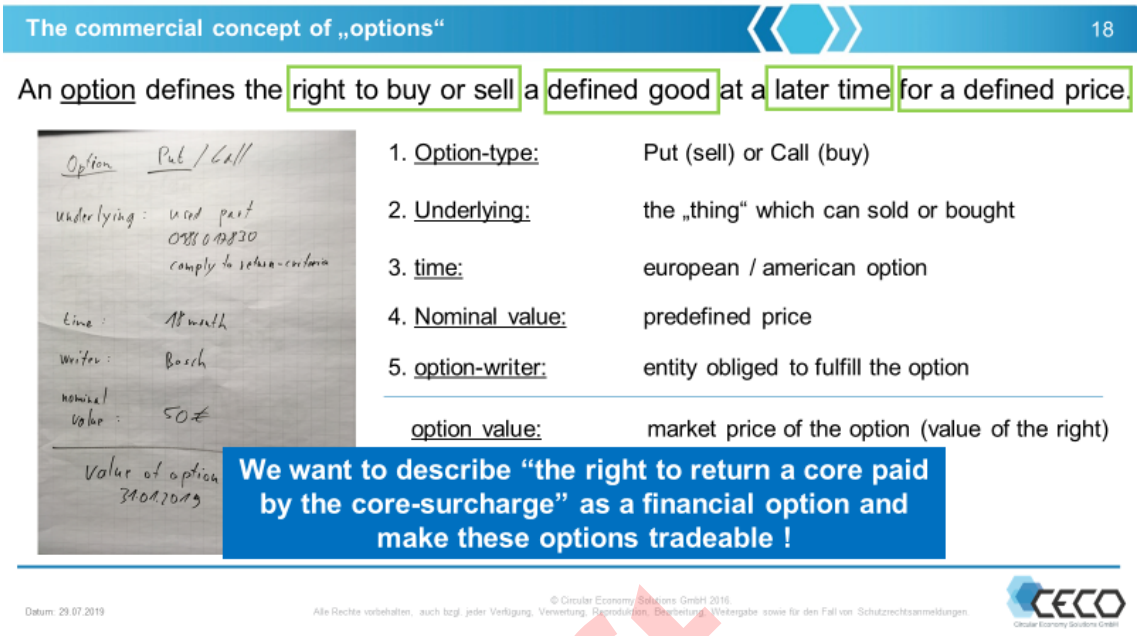


Figure 2: Definition of "option"

A very important thing to mention is the so called underlying. This is the artefact the option refers to. The challenge here is to describe the underlying in such precision that a stakeholder acquiring the option can be sure of what it is referring to, but at the same time allowing quite flexible, human-language-like descriptions which can nevertheless being interpreted by a machine. The underlying might not only define the “thing” or artefact as such, but also its state, conditions or location.

Two major roles associated with an option need to be defined and distinguished. The so called “option-writer” is the entity who is issuing/offering the contract represented by the option. The option-writer is responsible for the fulfillment of the contract and the obligations resulting from that. The so called “option-holder” is the current owner of the option. He/she is the one who could either sell and transfer an option to a new holder or execute the option towards the options-writer. Option-holder can change with the transfer of the option, whereas the option-writer stays always the same for a specific option as he/she is part of the definition of an option.

Changing of any of the characteristics of an option incl. the options-writer is not possible, as this would mean to do unilateral adaption to contract-conditions, which would influence the value of the option.

An option can only be terminated by the following events:

1. It is executed by the holder towards the writer. This means that the contract defined in the option is fulfilled.
2. It is not executed and time has passed over the due-date. In this case, the option cannot be executed anymore, is therefore useless and loses all its value.
3. Option-writer is buying back the option from free market and terminates it.

Example: Option granted with the sale of a Reman-part by remanufacturer:

1) Option-type: PUT, American.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

2) Underlying: (defined by option-writer: used part, starter-motor, Reference out of OE-Number cross-list from return-program matching to reman-sales-number e.g. 0001123456, compliant to core-acceptance criteria (exemplified in **Error! Reference source not found.**): e.g. Complete and not dismantled, no mechanical damage, no heavy corrosion)

Figure 3: Starter return criteria (Source: https://www.coremannet.com/return-criteria/)

3) Time: due-date for the option, e.g. 12 months after options issue-date
4) Option-writer: Remanufacturer e.g. Bosch (the entity granting the "right to return", responsible to fulfill the contract).
5) Nominal value: Predefined price the option-writer will pay to the option-holder in case he is executing the option (return a matching core to the remanufacturer), core-surcharge, e.g. 50€

2.2. Developments for Automotive Parts Demonstrator and Link to Circular Economy

2.2.1. Automotive Parts Demonstrator

The digital parts management-platform for the automotive demonstrator basically consists of two main systems:

1) An IT-system which is providing a central directory and history of all options and the functions to manage them. Additional to basic functions like to create, transfer and document options, it also provides the allocation of options to data referring from the return of physical items (e.g. core-return) to support the execution of options. It must also be capable to interconnect the execution of options along a chain of options via several actors. This system is also referred to as "clearinghouse".
2) Interfaces (machine2machine and machine2human) to connect and interlink the platform with its users.

The aims within ReCiPSS-automotive-demonstrator are

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

- to describe the "right to return" for cores to be used in automotive-reman as a commercial option,
- to develop clearinghouse-platform:
    - make the core-return-option an (digital) artefact which can be transferred and traded apart from a physical product,
    - generate the digital infrastructure to create, transfer and execute options,
    - connect this infrastructure with existing operational systems such as:
        - CoremanNet-System
        - Masterdata-systems
- standardized IT-protocol/ interface towards the market:
    - develop, implement and publish interfaces of ReCiPSS-digital-infrastructure to ERP-systems of stakeholders in automotive-aftermarket such as (traders, remanufacturers, service-providers),
    - connect with systems of 2 demonstrator-companies in France and Germany and handle in the demonstrator-phase options for 80.000 cores for them.

2.2.2. Expected impact to circular economy

There are several definitions of the concept of circular economy, as described in Deliverable D2.1: Defining the current baseline and the target/improved circular business models, Chapter 3.1.

The definition given by the European Commission, is the most relevant for our project "*In a circular economy the value of products and materials is maintained for as long as possible; waste and resource use are minimized, and resources are kept within the economy when a product has reached the end of its life, to be used again and again to create further value"* (EC, 2015, p. 1). [1]

The current business model of Bosch and other remanufacturers is a circular one: the cores are returned back from the market to the remanufacturing processes and after that again back to the customers.

The process of return is rather complicated, because the return chain is constituted by independent trade levels, which means that manufacturer (i.e. Bosch) loses ownership of the core and can't ensure the return of cores to remanufacturing.

Bosch is selling remanufactured parts under a special brand "Bosch eXchange", in order to indicate to everybody that for these parts a core-return is possible. Reverse logistics, identification and evaluation of the cores, sorting and storage as well as the surcharge balance (financial accounting) among the involved parties are operated by CoremanNet, a product of C-ECO.

Nevertheless, there is room to improve and even to enlarge that circular business. With increased transparency through the ReCiPSS platform, the overall acceptance for Remanufacturing and Circular Economy Business Models could rise – as the limited transparency has been identified as one of the main pain points for the wholesalers, leading to a probable surcharge deficit and a feeling of losing even more.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

In addition to the transparency topic, the stakeholders could enlarge their circular business by enabling them to introduce new business models. The trade channels beyond the remanufacturers will be able to trade not only the parts but also the options themselves. This will put them into the position of making additional turnover, not only by using up all of their core return options, but also by acquiring additional options for surplus cores.

The remanufacturers themselves will be able to introduce extra options for needed cores and sell them to any market participant owning the core without currently having the right to return that core. Today that process is only performed in between remanufacturers and core brokers – leaving the trade levels as the odd man out. In the future the trade levels can be involved which will not only be a value add for them, but it will also increase the probability for the remanufacturer to find the right cores he is looking for.

All in all, this allows for a better allocation of existing options to cores and, therefore, an increased core-return.

Additionally, the idea behind ReCiPSS opens the door for diverse service providers that can perform necessary or even partly unpopular process steps within CE. In case that a big remanufacture will be enabled to also make business with a small workshop or in the end with multiple workshops, this leads to the question of how to integrate such a big number of new suppliers into an organization with stringent workflows and complex ERP processes. No one will create a new supplier only to acquire a handful of cores, if the process to setup the new supplier costs many times higher than that. There will be a need for a financial clearing house that will bundle the activities from smaller workshops towards the remanufacturers serving as one supplier.

The core handling itself – sometimes seen as an unpopular activity due to its complexity and the contrast of handling dirty cores compared to selling shiny new or remanufactured parts – can be executed by an independent inspection centre. As the cores do no longer necessarily need to flow through all stakeholders, the independent but commonly accepted decision on core acceptance by a third party enables all levels of stakeholders to take a decision if the core handling can be outsourced. Bypassing trade level and single point core selection means amount of overall transportation will be reduced. Increased circulation of resources and lower transportation will bring environmental benefit in the form of lower carbon footprint, lower energy consumption and lower amount of wastes.

In a similar direction, since the business is no longer limited to only what happens between the initial seller and the initial buyer of a reman product, the system changes from a closed system to a more open one. As a vision, the cores should be returnable at any stakeholder when owning an option to return the core – similar to returnable bottles. The option created within the ReCiPSS platform should serve as a token that is accepted by all. Better information flow and single point core selection will make the process efficient and thereby core rejection rate will decrease. Another result will be the increase of return flow of cores.

Finally, all of the explanations given for the Automotive demonstrator, those benefits can be applied for other business-sectors within CE with similar processes. In the end, the type of part

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

that is handled as a core is not the decisive element. It is the tradeable option – and this can be adapted to all kinds of products within CE.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

3. Platform Architecture

In this section we will provide a brief summary of the development methodology and the software architecture used for the implementation of the IT platform.

3.1. Methodology of Development

For the project management process, we choose Agile Methodology, which is one of the adaptive software development approaches methodology. [2]

"Agile software development comprises various approaches to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s). It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change." [2]

From the start of T5.2, in month 7 (December 2018), thirteen Sprints of three weeks each were delivered until 8th of November.

The process involves breaking down the project into prioritized requirements and delivering each individually within an iterative cycle. Each iteration is reviewed and assessed by the development team and everyone involved. The insights gained from the assessment are used to determine the next step in development

In this way, we ensure changes can be made quicker and throughout the development process by having consistent evaluations and keeps the project transparent too by having regular consistent meetings that allow everyone involved to access the project data and progress.

.

The Agile Development Methodology ensures also a high quality of the product, due to the fact the testing is integrated in the development cycle.

3.2. Architecture

We value modularity in the architecture. This led us to adopt a service-oriented architecture commonly referred to as microservices architecture. Technologically this is supported by Linux containers running on a managed Kubernetes cluster in Microsoft Azure.

Error! Reference source not found. depicts the service architecture implemented for the platform. It comprises five microservices each responsible for a specific set of closely related tasks. The 'UI' and the 'Create Options API' form the interface to the public to facilitate the interaction of users and company ERP systems with the platform, both are accessible on the internet. The 'User Service' manages the identity and roles of the ReCiPSS users providing the login and authorization functionality. The 'Clearinghouse' is the heart of the platform. It

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

manages the creation, the transfer and the execution of options. Finally, the 'Customer ID Mapper' creates the connection of customer IDs, maintained by each company, to the ReCiPSS user IDs.

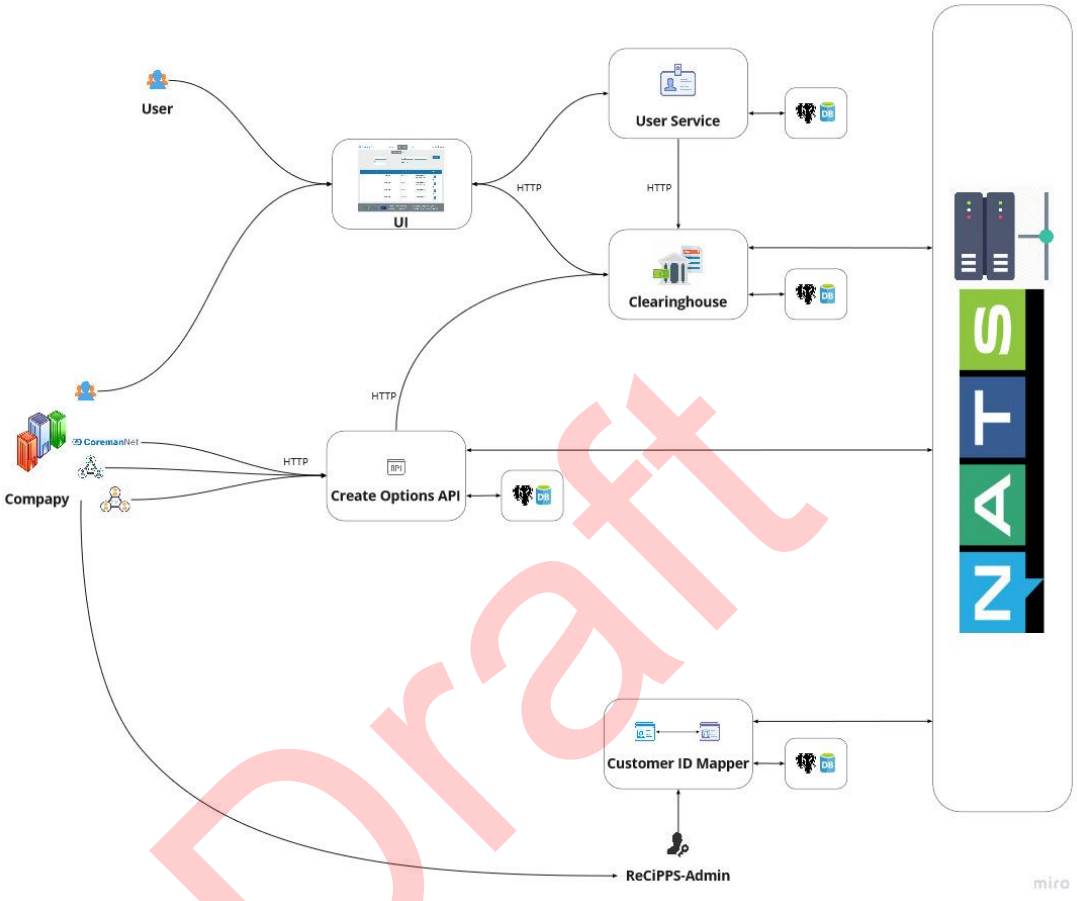


Figure 4: ReCiPSS architecture

As the platform is accessible online, the communication of the public interfaces runs on the HTTP protocol. Whereas the internal communication between the microservices uses NATS as a message queuing and streaming system, NATS also enables us to use an event-sourced architecture for each microservice. Each microservice stores all processed information in dedicated databases.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

3.3. Technologies used

For the web-based user interfaces of the platform we employed standard web technologies like HTML5 and CSS3, as well as more specific languages & frameworks like TypeScript, Angular 8.2.1. and NodeJS

Our backend system is implemented largely in Java 11 with the Spark web framework, but also experiment with NodeJS/ExpressJS-based microservices. Interservice communications are encoded as JSON and send via NATS Streaming and HTTP. For persistence of aggregated data we use Postgres and Redis.

We use platforms like Azure, Docker, Kubernetes, Bamboo. **Error! Reference source not found.** represents the technologies used as circular layers, starting with frontend, then backend, interservice communications and aggregated databases.

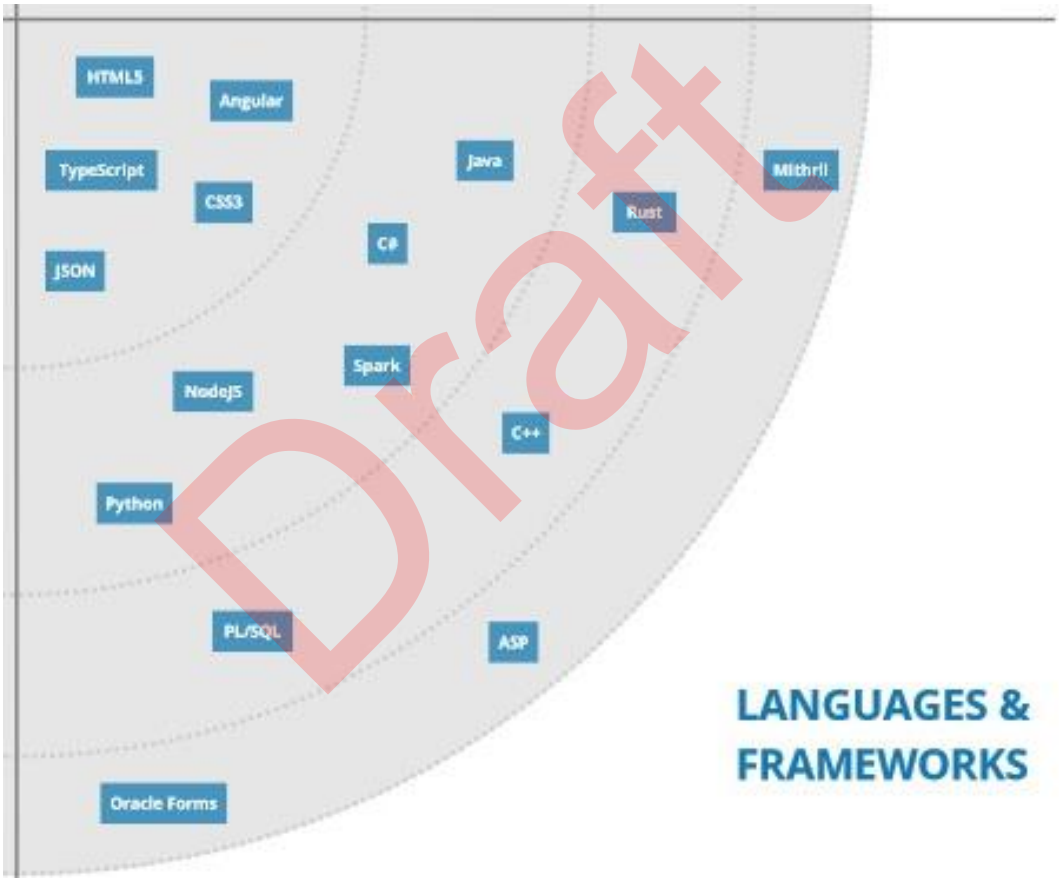


Figure 5: Technologies - languages and frameworks

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

3.3.1. Angular for user interface

We needed a dynamic client-side user interface framework, which is an object-oriented library to develop our frontend. It should:

- be compatible with most browsers
- be actively maintained
- support modern HTML5 web technologies
- let us communicate with the backend

Possible implementations that we considered were Angular, React, VueJS, Java Server Pages and Java Server Faces. We used Angular to develop our frontend based on:

- Angular's data binding (makes it easy to create components that update dynamically)
- Easy way to style the interface using the integration with its own materials and other materials (e.g. Google MDL, Bootstrap).
- Easy way to debug the frontend using a browser plug-in.
- It provides us with a component structure to follow to keep the code maintainable separating logic, layout and style. There are also services for non-UI functionality such as HTTP requests.
- There are packages available for internationalization.

Consequences:

- Angular produces rather large build artefacts, which increase initial load times for our users. We need to optimize our artefacts to control this.

3.3.2. Type script for user interface

We needed a language that compiles to JavaScript and specially used in client side. We need also a good structure design for all layers, with the possibility of using tooling, type safe and object-oriented programming.

We choose TypeScript for use the structure of the object-oriented programming language. Angular is written in TypeScript too. If necessary, JavaScript can also be used.

Consequences:

- TypeScript takes advantage of the structure of the object-oriented programming language. That keeps the code clean, scalable, consistent, and simple to debug.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

- It also provides advanced auto completion, navigation, and refactoring in development process.
- Take advantage of well-defined interfaces. And it is much easier to express the idea of an interface in a language that supports them.

3.3.3. Use Spark as a web framework

We needed a web framework to implement a microservice using Java. There are two fundamentally different approaches to this:

1.The batteries-included approach of established frameworks like Spring, Play or JavaEE. Usually they include many integration features for templates, persistence, etc. and prescribe a certain architectural approach such as Model-View-Controller.

2. Light-weight micro frameworks (i.e. Spark) make use of newer Java features to increase developer productivity. They do not prescribe a particular architecture and leave many decisions up to the development teams.

We used Spark as it is the most stable option in the micro framework space.

Consequences:

- As there is no architecture prescribed in the framework, we need to take care of related decisions ourselves.
- Furthermore, we need to choose libraries like persistence, etc. for other features as necessary that are not included.

3.3.4. Use PostgreSQL as database

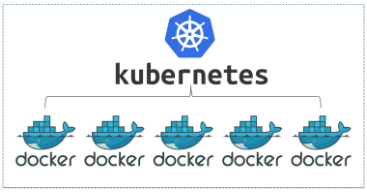
PostgreSQL is the world's most advanced open source database and yet is highly extensible with support for over 12 procedural languages, GIN and GIST Indexes, Spatial data support, and multiple NoSQL like features for document (JSON) or key-value based applications.

Some major advantages regarding the use of PostgreSQL:

- Extensible - The source code is available to all at no charge. This allows the freedom to use, modify, and implement it as per business needs.
- Cross platform – It's available for almost every brand of Unix and Windows versions 8.0 and above natively.
- Designed for high volume environments –it uses a multiple row data storage strategy called MVCC to make PostgreSQL extremely responsive in high volume environments.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

- PostgreSQL supports ACID (Atomicity, Consistency, Isolation, Durability).
- Its supports JSON data.



3.3.5. Use Docker and Kubernetes

We needed a good way to bundle and run the application. Containers have relaxed isolation properties to share the Operating System (OS) among the applications. Therefore, containers are considered lightweight. A container has its own file system, CPU, memory, process space, and more. As they are decoupled from the underlying infrastructure, they are portable across clouds and OS distributions.

Docker is a standalone software to run containerized applications. Containerization is an approach of running applications on an OS such that the application is isolated from the rest of the system. You create an illusion for your application that it is getting its very own OS instance, although there may be other containers running on same system. Docker enables us to run, create and manage containers on a single operating system.

Kubernetes is a platform for managing containerised services. It is most commonly used with Docker managed containers, although it doesn't strictly depend on it. It allows the container's provisioning, networking, load-balancing, security and scaling across all these nodes to be done from a single command line or dashboard. A collection of nodes that is managed by a single Kubernetes instance is referred to as a Kubernetes cluster.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

4. Summary of the functionalities

This section summarises the main functionalities that the Automotive part data exchange platform includes.

4.1. Standard Functionalities / no business functionalities

- Login / logout procedure
- Readable account names: email address

The login procedure available on this platform at this moment is the standard one – using a username and a password. There are two categories of users: regular users and administrators. The regular users are dealing with main functionalities of the applications – options, transfers, containers, payments, etc., and the administrators are dealing with users management, i.e. users creation and companies – users mapping.

4.2. View options

There are different possible views on options: the holder and the writer. The option writer initially creates the option and is the trading to which the underlying can be sold or from which it can bought. At any given time, the option is the property of some legal entity called the option holder.

- **Holder & Writer**
- **Sorting functionality**
- **Filter functionalities**
- **Expiry of options**

4.3. Transfer options

The option holder could transfer his options to other users in 2 ways. He could transfer a single option or he could create a container: combine all the option in my wallet to a container to combine a lot/batch which can be transferred as bulk without having the need to process every single option.

- **Transfer single option**
- **Transfer multiple options**
- **Transfer container**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

4.4. Create options

- Create option via platform
- Create option via interface

The above mentioned features are described in the next chapter- 5. Main components.

The following functionalities are planned to be developed and added to the platform until month 24:

- User Management - Company Aggregation:
  For the time being, users of the platform act as individuals. This will be changed as we will implement the entities of companies. As soon as this is done, all users will be assigned to a company to enable the overview of and the transfer of options not in between individuals but in between companies.
  In addition, advanced user profiles will be created to grant dedicated roles depending on the function of a user within a company. Only some user will be allowed to transfer cores, but most users of a company will be able to have an overview on the surcharge options.

- Grouping functionalities for option views:
  Currently, the options are displayed one by one. For a better usability the options will be grouped in case they have the same attributes.

- Underlying Definition:
  To enable the platform to understand the different logics of the diverse suppliers, a logic that goes beyond the pure part number logic needs to be applicable. Therefore, the platform will consider the entity of an underlying as the combination of multiple variations of core definitions (example 1: part number 4711, colour green, status good part; example 2: clutch disc, diameter between 350 and 430 mm, status good part).

- Connectors for interfaces with other systems:
  As a start the ERP systems of the demonstrators will be connected. This will be done with dedicated ERP-related connectors. One of those will be SAP R/3 which is supposed to be used by a lot of potential users. During the demonstration phase additional ERP systems and connectors might be tackled.

Following functionalities will be implemented as the part of the automotive demonstration phase when the platform will be deployed in the actual use environment:

- Market Value of Option:
  As the nominal value is a fixed characteristic within the option, for the transfer another value – the so-called market-value – needs to be implemented. This value will be the one that is actually paid for obtaining the option. Though the value itself will be

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

negotiated outside of the platform, it needs to be incorporated into the platform as an entity to enable automated payment processes that are linked to the transfers.

- Payment Process:
  In parallel to the incorporation of the market value, the payment process needs to be triggered in the moment that a transfer is made based on the market value or the price for a container. In case the user is connected via defined interface, the clearing house can initiate a transmission to the ERP-system of the user to create a payment related transaction there. In case a user might not be connected via interface, alterative payment methods need to be taken into consideration.

- Reporting:
  Depending on the experience and the input of the users of the demonstrators, more sophisticated reports will be developed and integrated.

- Allocate options / Execute options / Allocate cores to options via interface:
  When cores are identified and evaluated, before sorting them into box that will be sent to the remanufacturer, the system needs to check to which remanufacturer it should be sent. The information will come from the platform, as all the options are managed from here. This process requires a fast response time between the platform and any IT supported inspection software. For the demonstration phase, the C-ECO hosted inspection tool will be connected to the platform.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

5. Main components

This section describes the functionalities that are currently part of the platform.

To login in the platform the user first will create an account and a password. This allows the user to enter automotive parts into the database (by creating an option) with certain specifications (name, price, expiration date, number of products available of the same type).

After the introduction of the products, they will be found in ‘Surcharge options’, where they can be viewed in a table, along with all the information introduced into the platform. With the ‘transfer button’, several products of the same or different type can be transferred to 'Transfer wallet' page, where a package will be created, which will contain the previously selected products. This package is a container that will receive a user name.

After the package creation, a pop-up will be displayed that presents a summary of the products in the created package. Then these created packages can be viewed in 'Containers', where you can see the information about the name of the package, the price, the number of existing products in that package, and a few buttons to see information about the package, to transfer the package to other users, unpack the package and share the package with other users.

5.1. Login/logout procedure

The login page (shown in

Figure 6: Login page

) contains the following information: user e-mail address field, user password field, password recovery option, and sign in button.

In the login box, enter the email address and password designated when your CoremanNet user account was created.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

Figure 6: Login page

To logout out from the platform you have to click top right on 'Log Out'.

5.2. Create new user

To add a new user use the option from top right 

, as is shown in

Figure 7: Create a new user

: Create a new user.

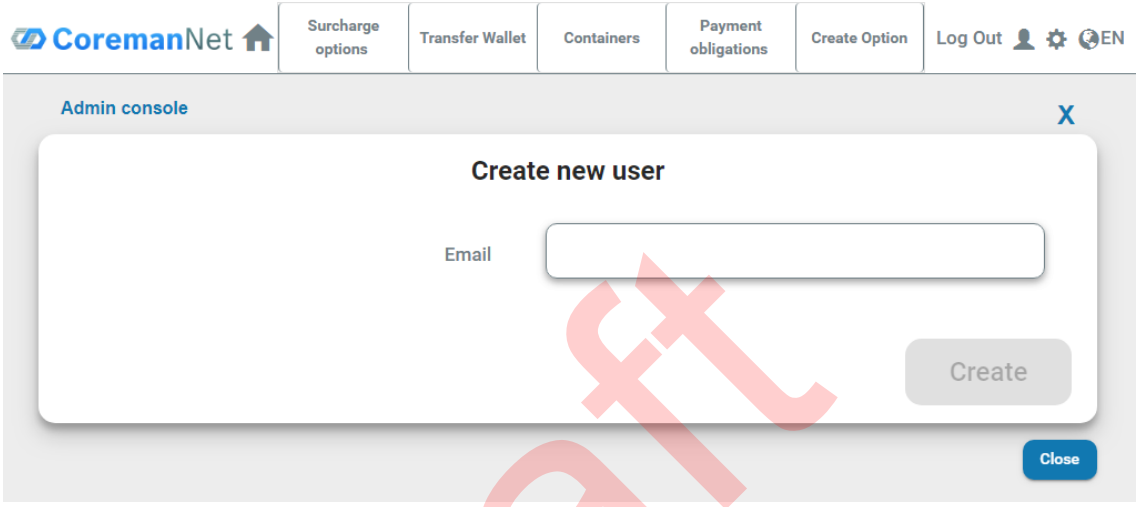


Figure 7: Create a new user

Currently there are only two different user types – administrator and standard user. More complex user and role management will be implemented at a later stage.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

5.3. Welcome page or platform dashboard

Upon successfully logging into the RecipSS platform, a dashboard page will appear
(

Figure 8: ReCiPSS platform welcome page

) with the following content:

- Surcharge options (Analyse and select options):
  This is the view where the user can see all the options that he holds. As a standard case, the platform created those options for the user in the moment he was purchasing a reman product from his supplier (writer of the option).

- Transfer wallet (Transfer selected options):
  When the user decides to trade options or to at least offer options to be traded, the selected options will be moved to the transfer wallet for further processing. In the wallet the user can transfer options to another user or create a container of multiple options for further processing.

- Containers (View and trade bundled options):
  Containers itself can also be transferred. Whilst trading options from the wallet means that every single option is transferred with an own value to another user, the transfer

of a container only requires a single value for the whole container containing multiple options.
Containers can also be shared with other users which give them the possibility to view the details of the contained options upfront of trading for them.

- Payment obligations (Track options you created):
  The view is similar to the view within the “surcharge options”, but the options those payment obligations correspond to the options that the user has written or created. In a standard case, the platform created those options for the user in the moment he was selling a reman product to his customer (holder of the option).

- Create option (Create new options):
  Options could either be created automatically by transmitting sales / purchases via a defined interface or they could be created manually within the platform.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

Figure 8: ReCiPSS platform welcome page

5.4. Create option

After login in theReciPSS platform, the first time we will have to create a new option using

"Create option" button  from the Dashboard page, as seen in **Error! Reference source not found.**: Creation of a new option.

The form should be filled in with following info:

- Underlying / Part number:
  Is a valid product number of the parts that will be become new options

- Nominal option value:
  Is the price of each option newly created. This price is the amount of money that the writer is willing to give for every core that they will receive if it fulfils the criteria (underline).

- Expiration date (option to auto select 12 months or 18 months or select a date from the calendar):

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

Is the period while the option writer waits to receive cores. After this expiration date the option is not valid anymore (and if someone returns a core after this date, the core will not be paid back.)
To set expiration date you can use the option to auto select 12 months or 18 months from current day or you can use calendar picker.

- Quantity:
is the amount of options that the writer wants to issue.

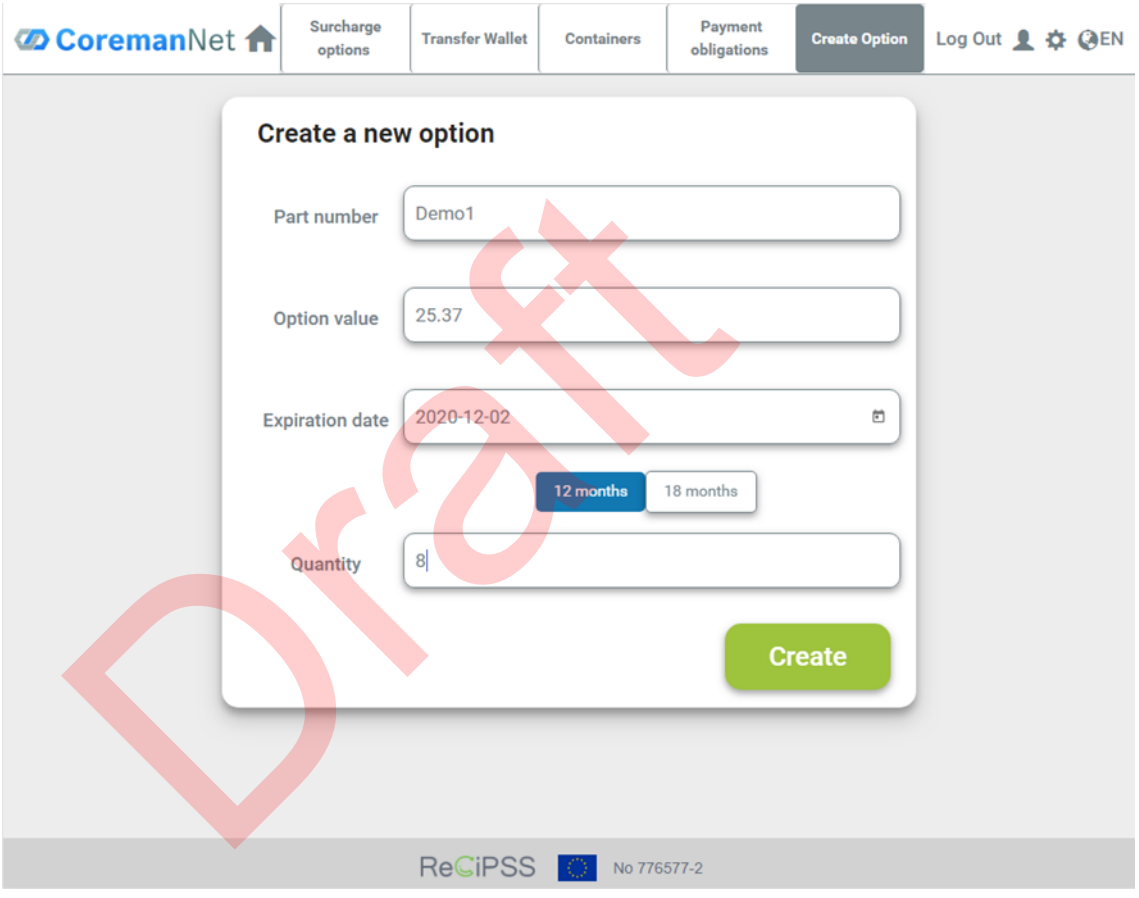


Figure 9: Creation of a new option

5.5. Surcharge option

In this view (see **Error! Reference source not found.**: Surcharge option view) the user can see all the options he holds. Holding an option means to have the right to return a core with the corresponding option for a certain value. This right can be executed towards the option writer.

As default 10 records will be listed with the option to change to 5 or 20 records per page.

In top of the list you have the possibility to filter data by value, expiry date or writer name.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

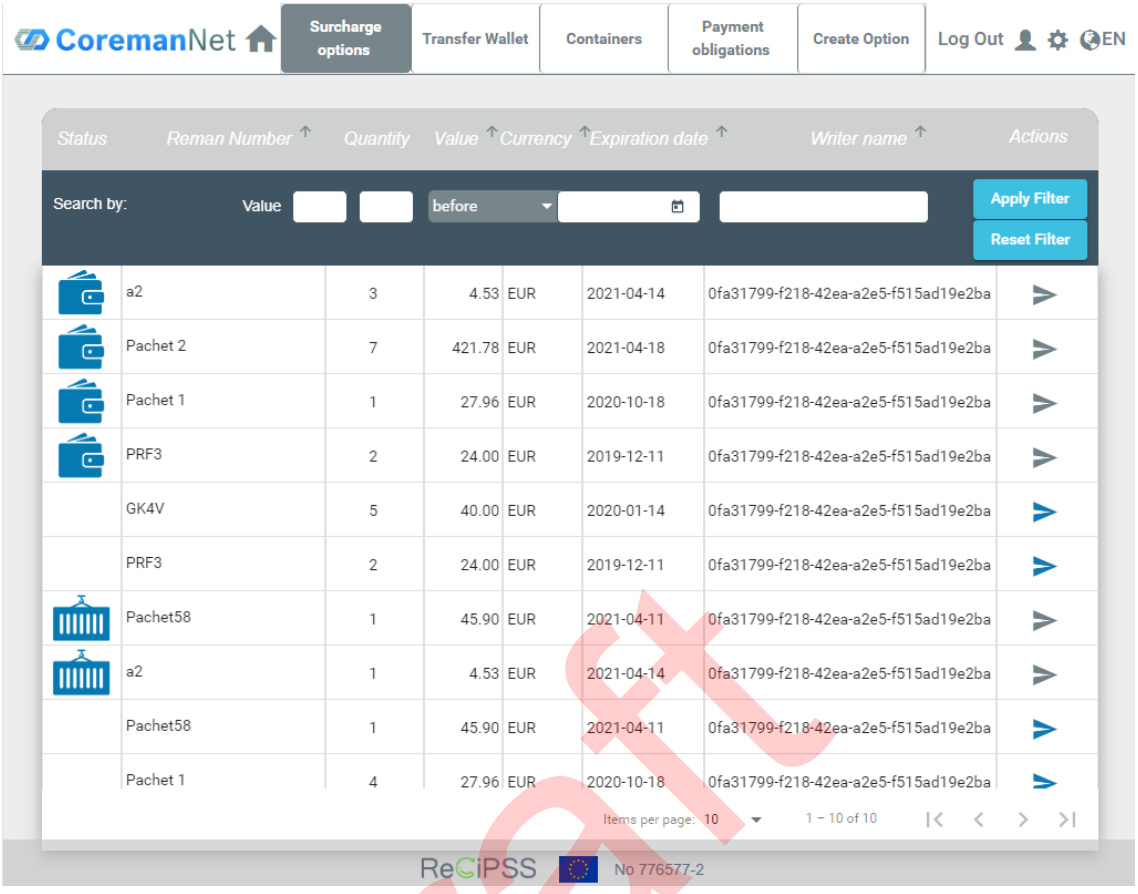


Figure 10: Surcharge option view

The following analysis can be done based on that view:

- sort and filter options by option writer to see how much surcharge value is open (and use this information e.g. in negotiations with supplier)
- sort and filter options by expiry date to see when the next options will expiry (and consequently focus on the ones that will expire soon)
- sort and filter options by value to see the most valuable options (and consequently focus on the most valuable ones)

One use case could be that the holder of the option has analysed his options and finds that some options will expire soon. The user knows that he will not be able to execute that options as he has no cores to do so. In that case he can still try to sell the options to another wholesaler or a corebroker who might have such cores but is maybe missing the return options.

The user could therefore select the corresponding options “Reman Number” you want to send in the wallet and press ➤ and add them to the wallet for further processing.

5.6. Transfer wallet

The selected options from the “surcharge options” view will appear in the “transfer wallet” view: see **Error! Reference source not found.** bellow.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

Here we have option to transfer the option(s) directly to another user, to create a container or to clear wallet.

Clearing the wallet means that the options from the wallet will neither be transferred nor put to a container. Those options will remain in the portfolio without being visible in the transfer wallet any longer.

If the user wants to transfer the options from the “transfer wallet” to another user, this means that each option will be offered or sold as an individual position with an individual price for each option.

The email address from the right list has to be chosen and there is also the possibility to search for an e-mail in the search field in case the relevant user is not visible at first glance. As soon as an e-mail address is selected, the transfer button will be activated. As soon as the button “Transfer” will be clicked, the options will be transferred to the selected user. Consequently, those options will disappear from the “transfer wallet” and from the user´s portfolio.

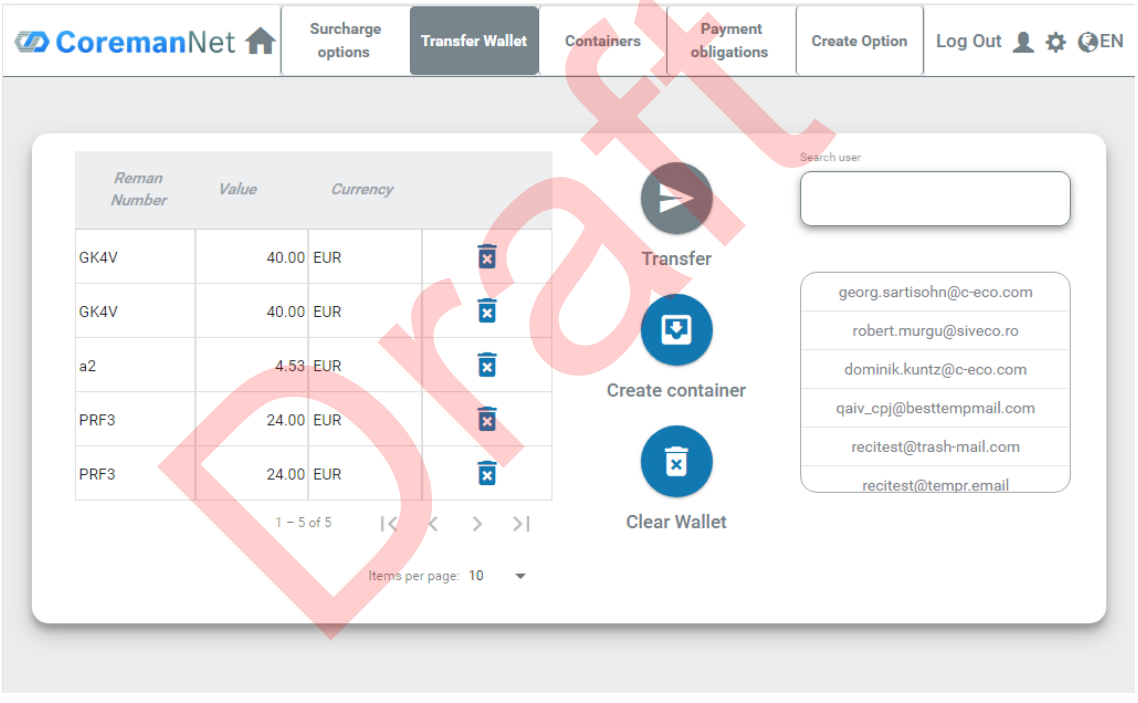


Figure 11: Transfer wallet

If the user wants to create a container, this means that he wants to put all those selected options together and offer or sell them as a single position and with one price for the whole container.

To create a container, you must to click on the button:  and then a pop-up will appear (as in **Error! Reference source not found.**: Creation of a container). Now you have to put a name to container and confirm it. After you confirm the container name, another pop-up will appear where you can see the container summary (information about the contents of the container created and information about the products that exist in this container- as in Figure 13: Container summary).

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

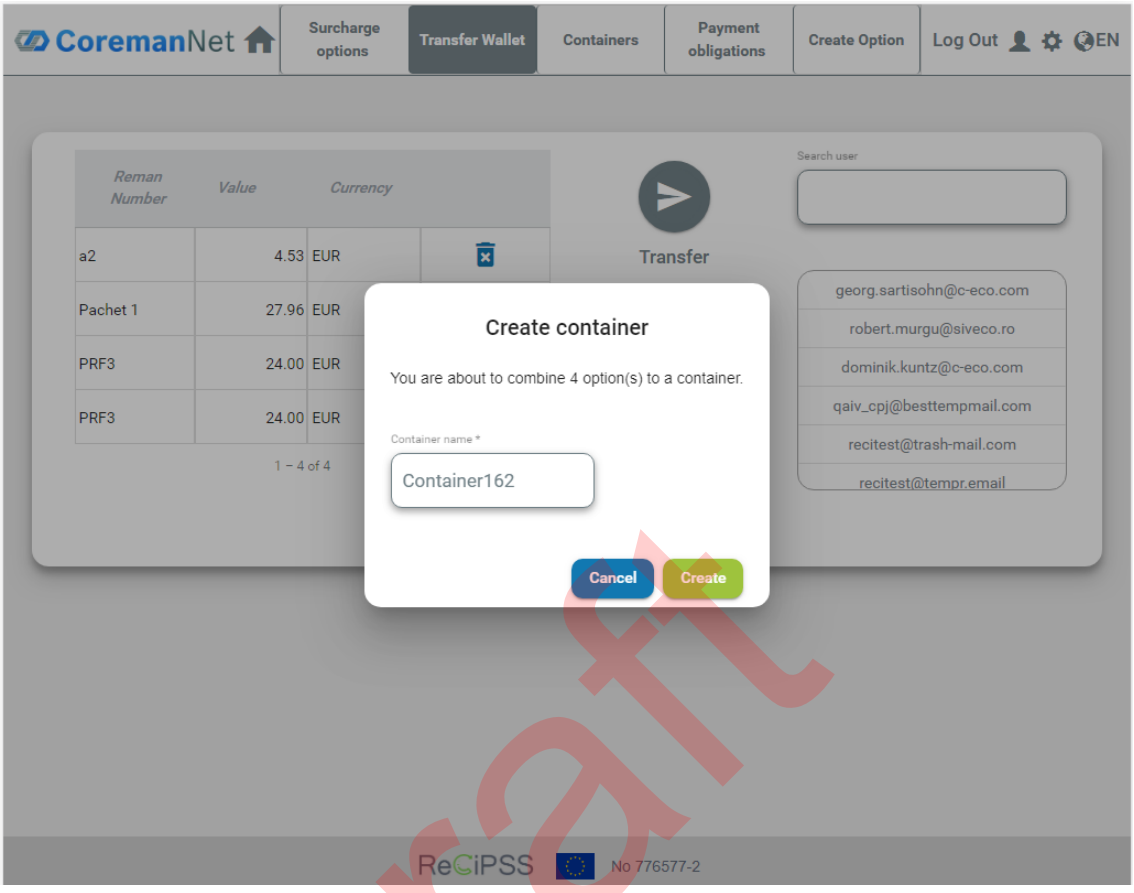


Figure 12: Creation of a container

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

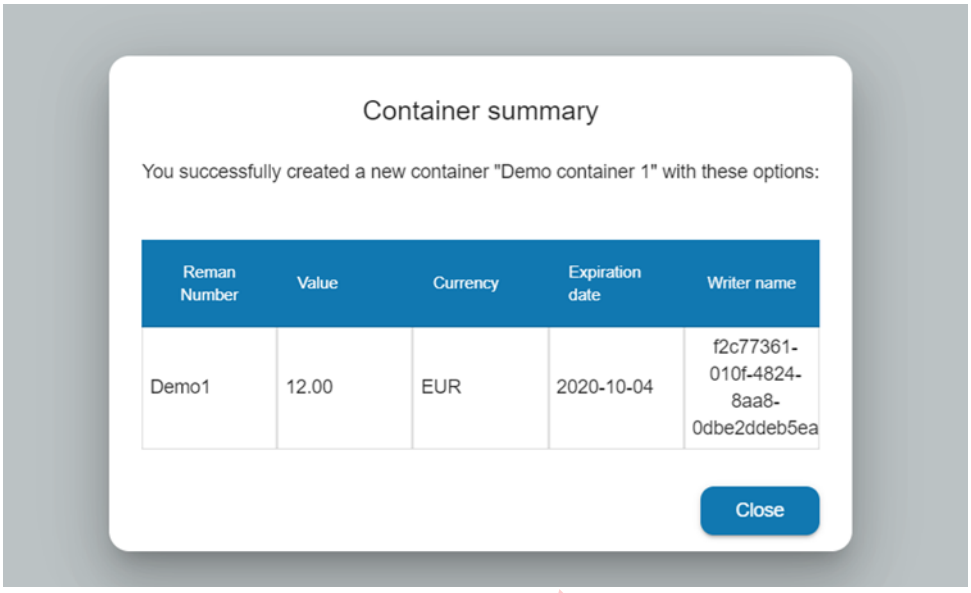


Figure 13: Container summary

5.7. Containers

The containers are mostly used in the business relationships with core brokers.

In this section you will find the explanations of the further processing of the containers that can be viewed, transferred, unpacked, shared (**Error! Reference source not found.**: Containers; List of all the containers that the user has created)

In the menu “containers” the user will not only find the containers that he created by himself, but in “my shared containers” he will also find the containers that were created by other user and that were shared with the user.

In the overview, the name of the container, the number of options and the sum of the nominal value is displayed. For shared containers also the holder of the options with that shared container is shown.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

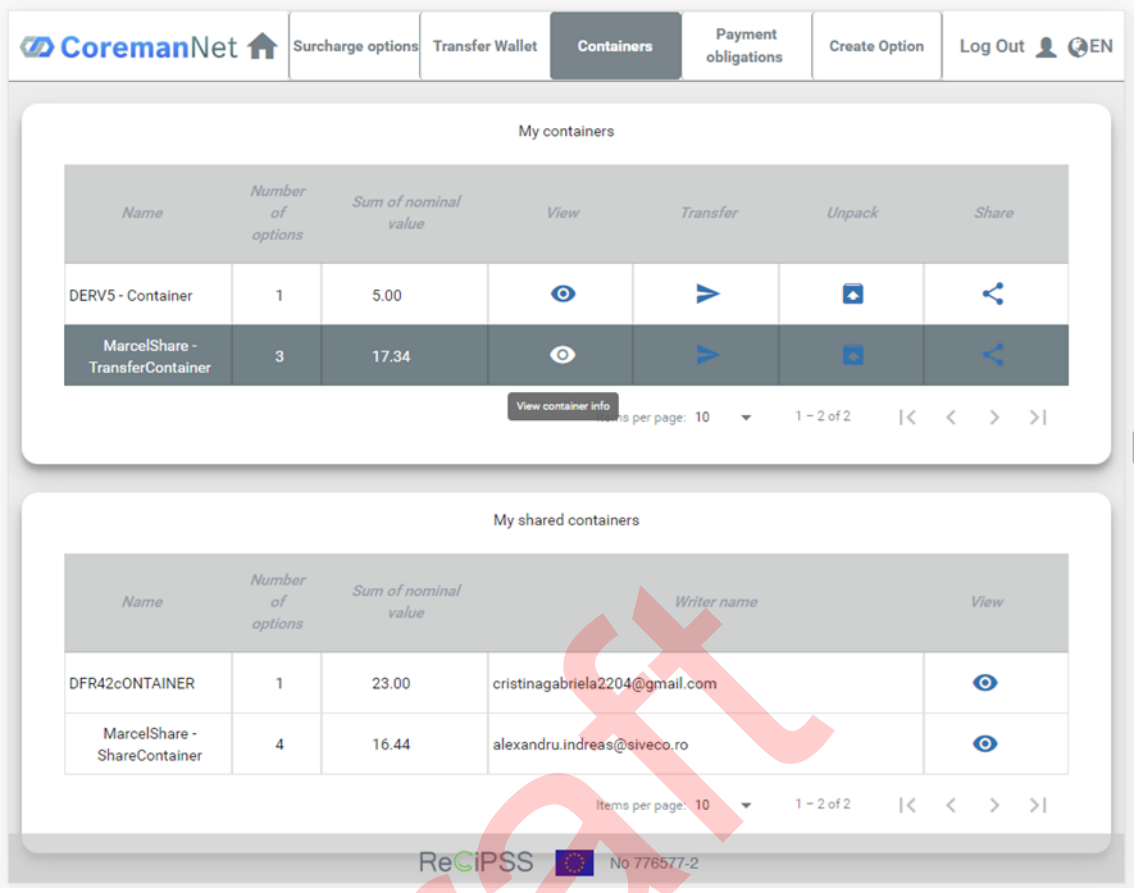


Figure 14: Containers; List of all the containers that the user has created

5.7.1. Container view

Both for own containers and for shared containers the user has the possibility to view the details of the container – **Error! Reference source not found.**: View of the information of a selected container.

If you click on the  button, a pop-up window will appear where you can see the detailed information of each option within the container. The details as reman number, nominal value of the option, expiry and writer name are presented in a similar way as in “surcharge options”.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

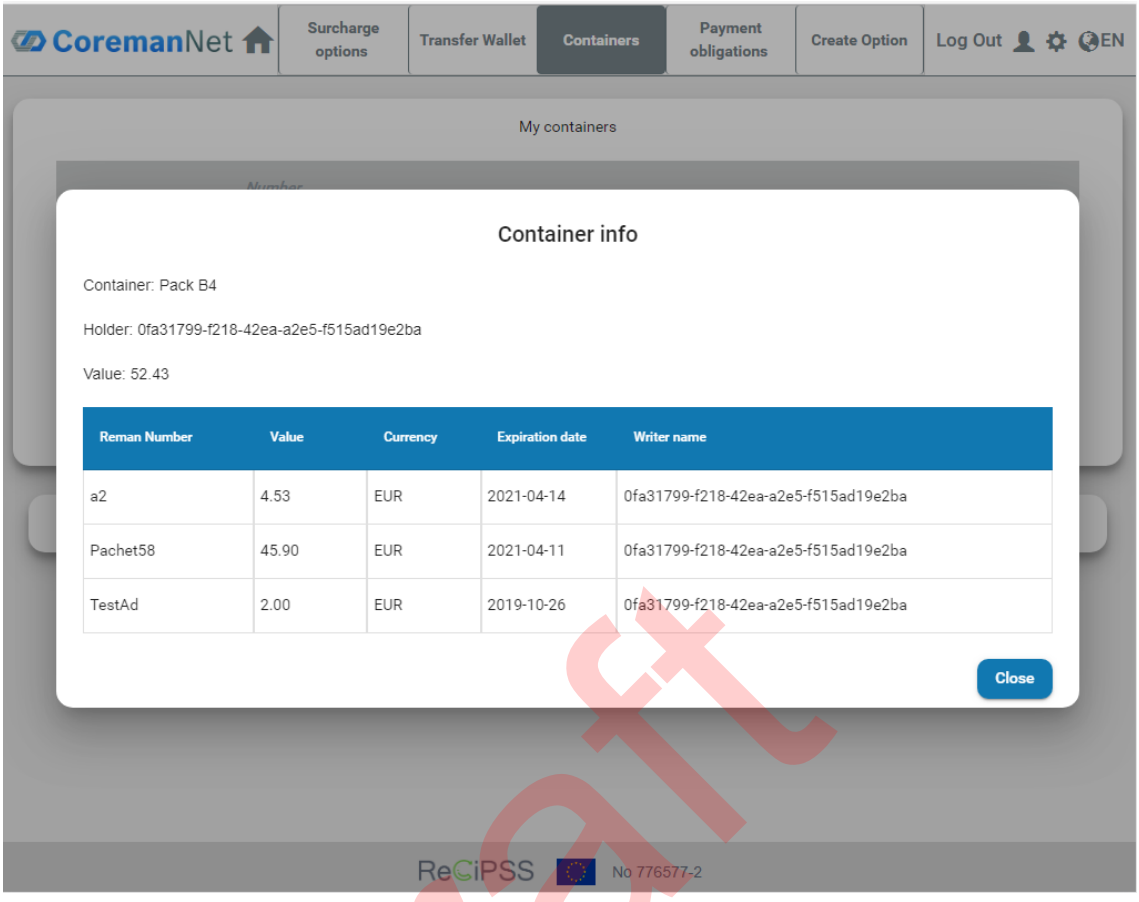


Figure 15: View of the information of a selected container

5.7.2. Container Transfer

If a container will finally be transferred, the user must click on the  button. A pop-up (see **Error! Reference source not found.**: Transfer Container) will appear where the user needs to be selected. In case the user is not visible at first glance, the search field can be used to find the relevant user. When a user is selected, the “confirm” button will be activated.

When the transfer is confirmed, the container is transferred to the selected user, the container will disappear from the container list and all options from the container will disappear from the portfolio.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

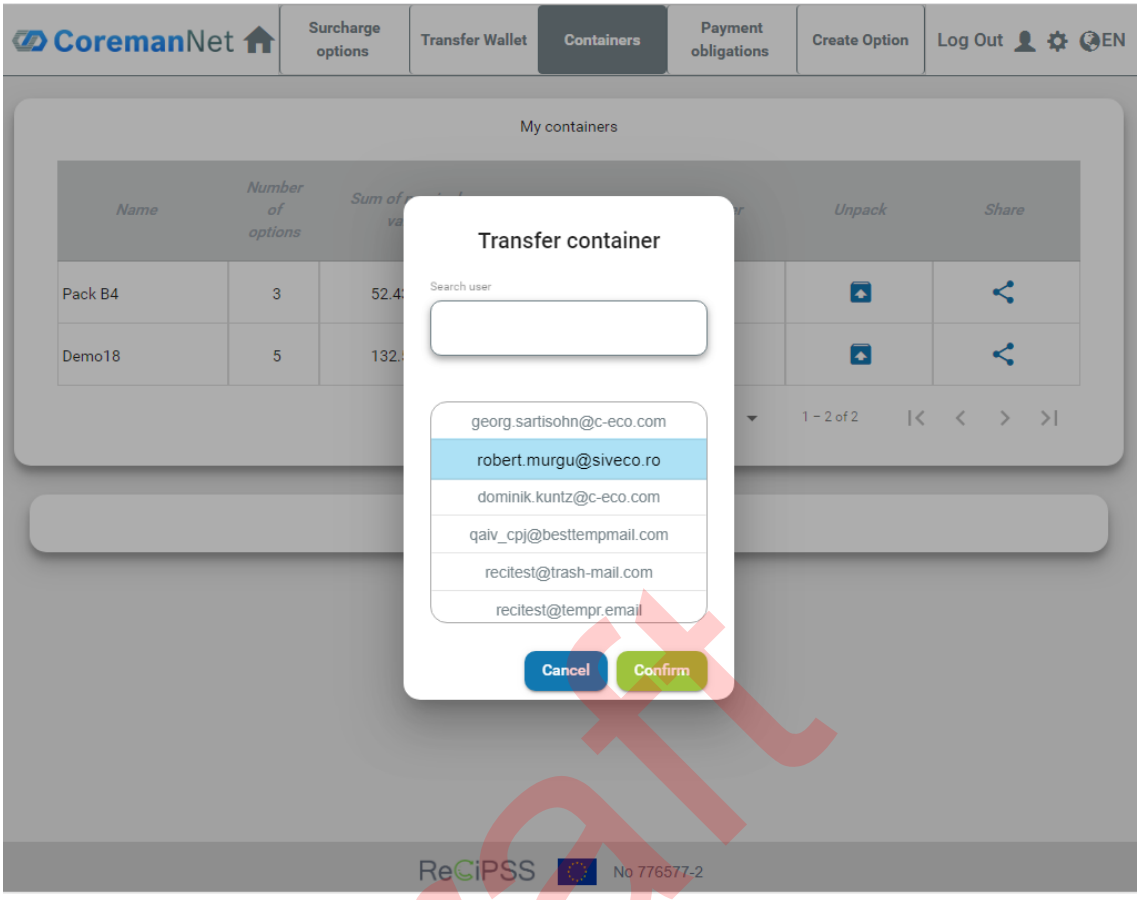


Figure 16: Transfer container

5.7.3. Container Unpack

Unpacking a container means that the container will neither be transferred nor shared. The container will disappear from the container list and all options will remain in the portfolio.

If you want to unpack a container, you must click on the button. After that a pop-up window (see **Error! Reference source not found.**: Unpack of a container) will appear where you can choose where the options should be moved. They could be moved into the transfer wallet or into the option-depot. In the transfer wallet the options could either be transferred or packed into another container.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

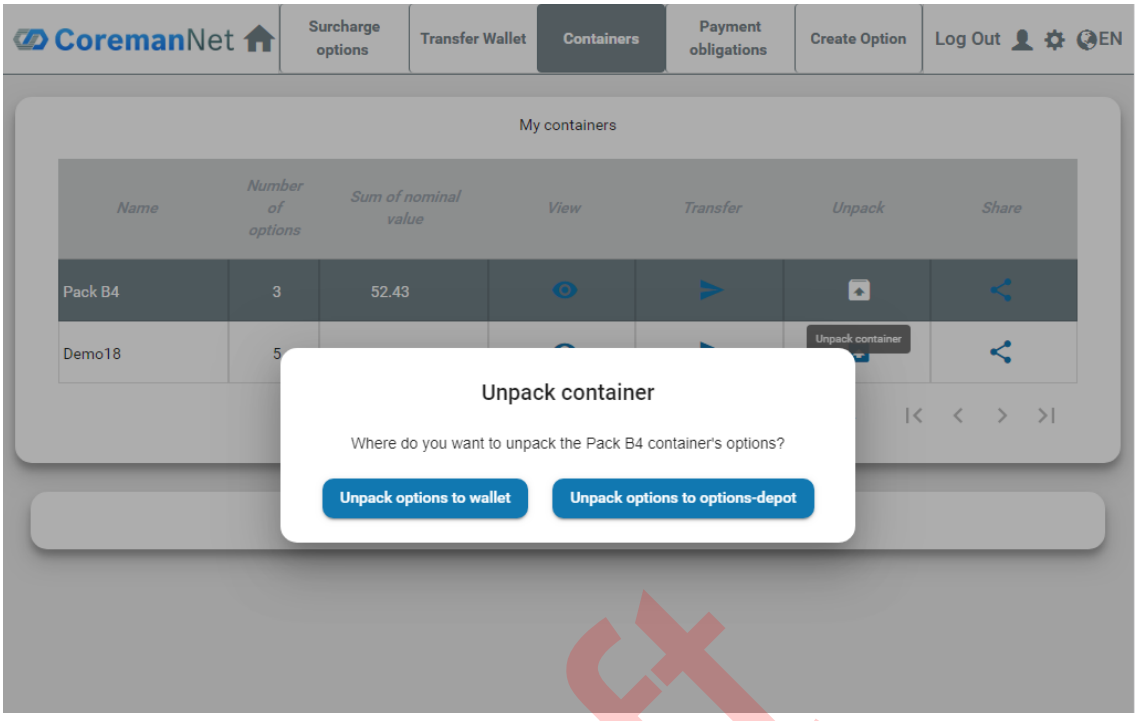


Figure 17: Unpack a container

5.7.4. Container Share

Upfront transferring a container to another user, the containers could be shared. This means that the packed container can be viewed and analysed by other users. With that functionality the potential purchaser of the options could decide if the options in the container are valuable and if so, to which value.

If you want to share a container with other users, you must click on the button. After that a pop-up window will appear where you have to choose one or several users you want to share with. As soon as a user is chosen, the “share” button is activated. When sharing is confirmed, the container can be viewed by the selected user(s).

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

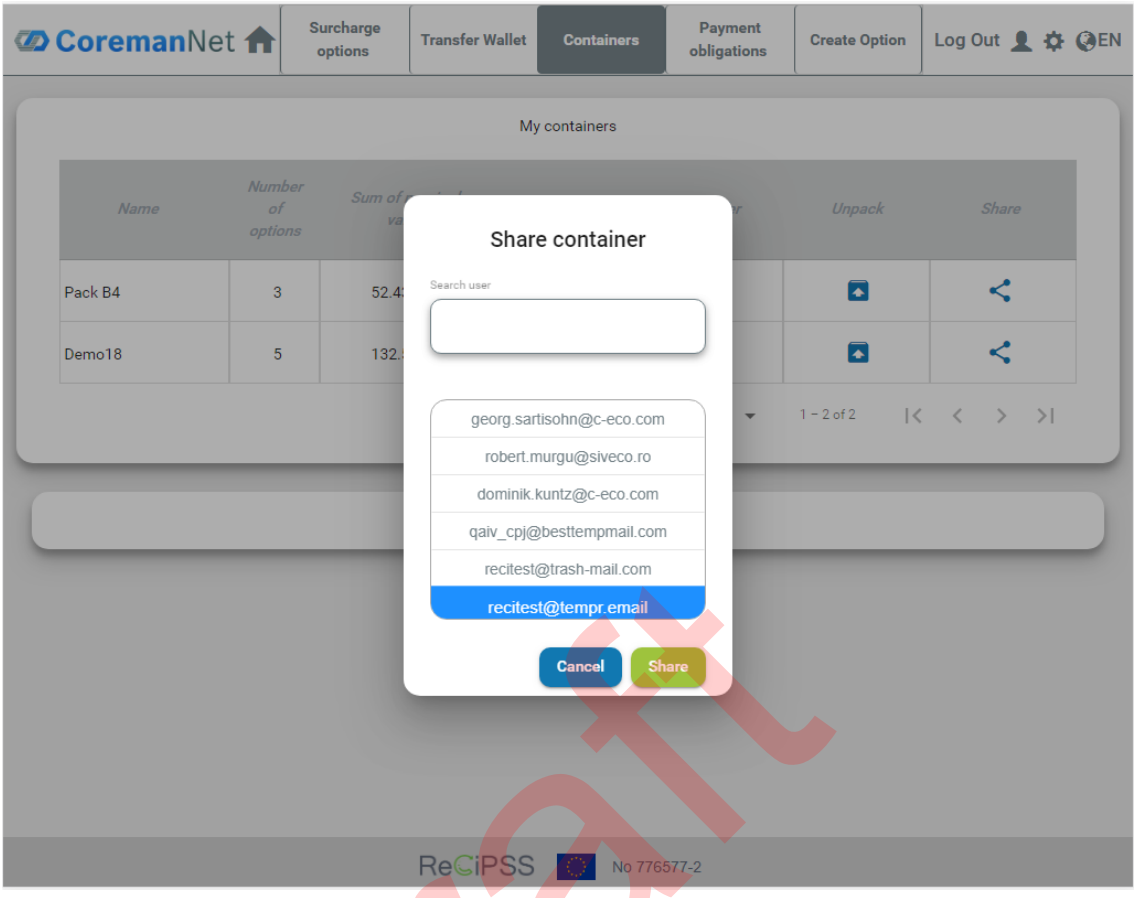


Figure 18: Share a container with another user

5.8. Payment obligations

Compared to the "surcharge options" where all options can be viewed that a user holds, in the payment options all options are listed that a user has written. Writing an options means to have the obligation to pay a value in case such an option will be executed by the option holder (returning a core with the corresponding option).

As default ten records (see Figure 18) will be listed with the option to change to five or twenty records per page.

In top of the list you have the possibility to filter data by:

- value
- expiry date or
- holder name.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

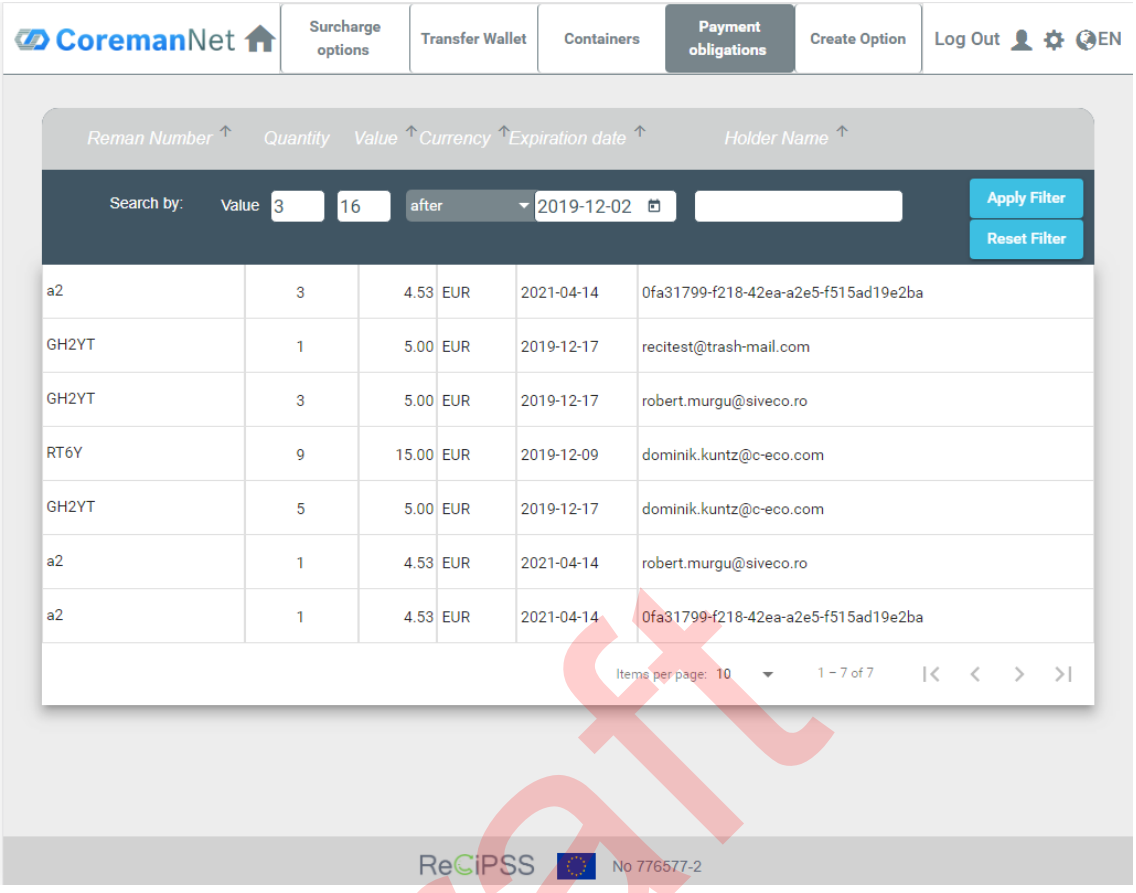


Figure 19: Payment obligation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2

6. Conclusions

This report presents the details of automotive part data exchange platform demonstrator, which is due in month 18 (November 2019).

Until now, we implemented the standard login, logout functionalities. We created view options (table that contains sorting functionality, filter functionalities), expiry date (implemented for options).

We created the “transfer option” functionality and implemented the “create options” one via platform and via interface.

For the coming months, we will implement the User Management - Company Aggregation, grouping functionalities for options view table, define logic for generic descriptions of underlying, create descriptions of underlying which can be repeatedly used for options, make underlying-descriptions available to platform-user, connecting interfaces with other systems, allocate and execute options, market value of options and payment process. At the end of its development the part data management platform will help the involved stakeholders to change current patterns of reverse supply chains behaviour in returns of automobile spare parts and to improve this process. The optimization of the reverse logistics processes could bring reduction of costs (for warehousing, transport, handling processes) for the individual trade levels. At another level it also can help to reduce pollution (reducing transport) and maximize the usage rate of cores for remanufacturing (and so to reduce raw material extraction, transport and processing).

The automotive part data exchange platform, after deployment of the integrated solution in the automotive parts demonstrator, will simplify tracking and tracing the flow of cores and reman parts, their status, the managing relationships between the users and providers of platform, contributing to a higher transparency and higher speed of reverse logistics processes. Additionally, it is expected that by the use of the automotive part data exchange platform, the return flow of cores will be increased, and better information flow and single point core selection will make the process efficient and thereby core rejection rate will decrease. As a result of the increased amount of returned cores less raw materials will be needed in the future. With the single point core selection at the inspection centres the overall transportations will be reduced. Less raw material and lower transportation will be reflected also in environmental terms, with lower carbon footprint, lower energy consumption and less waste.

The Agile methodology applied in fulfilling the objectives of task T5.2 was proven a good one and should be continued in the same manner until the end of the project. It would be replicated for other work packages, i.e. WP6 - White Goods Demonstrator and WP7 - Automotive Parts Demonstrator.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 776577-2

7. References

[1] EC (European Commission) (2015). Circular Economy Package: Questions and Answers.

[2] https://en.wikipedia.org/wiki/Agile_software_development

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776577-2