



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

TREBALL FI DE GRAU

Grau en Enginyeria de l'Energia

**DISSENY D'UNA APLICACIÓ DESCENTRALITZADA PER A LA
GESTIÓ D'UNA COMUNITAT ENERGÈTICA LOCAL**



Memòria i Annexos

Autor: Iu Perramon Barris
Director: Àlex Alonso Traveset
Co-Director: M^a Helena Martín Cañadas
Convocatòria: Gener 2022

Resum

En aquest treball es realitza el desenvolupament dels “*smart-contracts*” o contractes intel·ligents com a base d’una aplicació descentralitzada o Dapp per a la gestió energètica i la seva facturació d’una comunitat energètica. Per dur-lo a terme, s’ha realitzat un anàlisi del marc normatiu espanyol sobre l’autoconsum col·lectiu i les comunitats energètiques. També s’estudia i compara els ecosistemes “*blockchain*” o cadena de blocs on desenvolupar aquests “*smart-contracts*” d’entre els principals dins el sector. Dins aquest programari desenvolupat s’estableix una estratègia de repartiment en funció de coeficients dinàmics *ex post* que actualment no estan contemplats dins la normativa. Més endavant s’explica detalladament el codi utilitzat pel desenvolupament d’aquest programari i es realitza una simulació d’aquest. Conjuntament s’exposen dos escenaris diferents, però partint de les mateixes dades base, per poder comparar els diferents escenaris plantejats. D’aquesta forma s’observa que la implementació d’aquest programari i estratègia de distribució aporten tant beneficis econòmics i socials com mediambientals.

Resumen

En este trabajo se realiza el desarrollo de los “smart-contracts” o contratos inteligentes como base de una aplicación descentralizada o Dapp para la gestión energética y su facturación de una comunidad energética. Para llevarlo a cabo, se ha realizado un análisis del marco normativo español sobre el autoconsumo colectivo y las comunidades energéticas. También se estudia y compara los ecosistemas “blockchain” o de cadena de bloques donde desarrollar estos “smart-contracts” entre los principales en el sector. Dentro de este software desarrollado se establece una estrategia de reparto en función de coeficientes dinámicos ex post que actualmente no están contemplados en la normativa. Más adelante se explica detalladamente el código utilizado para el desarrollo de este software y se realiza una simulación del mismo. Conjuntamente se exponen dos escenarios diferentes, pero partiendo de los mismos datos base, para poder comparar los distintos escenarios planteados. De esta forma se observa que la implementación de este software y estrategia de distribución aporta tanto beneficios económicos y sociales como medioambientales.

Abstract

In this work, the development of “smart-contracts” is carried out as the basis of a decentralized application or Dapp for energy management and billing of an energy community. To implement this, an analysis of the Spanish regulatory framework on collective self-consumption and energy communities has been carried out. Blockchain ecosystems are also studied and compared to develop these "smart-contracts" among the main ones in the sector. Within this developed software, a distribution strategy is established based on dynamic ex-post coefficients that are not currently contemplated in the regulations. The code used for the development of this software is explained in detail later on and a simulation of it is carried out. Additionally, two different scenarios are exposed, but starting from the same base data, in order to be able to compare the different scenarios proposed. In this way, it is observed that the implementation of this software and distribution strategy brings both economic, social and environmental benefits.



Glossari

Conceptes

- “Address”: Element que emmagatzema qualsevol valor d’una adreça corresponent a la “blockchain”.(1)
- “Array”: Estructura d’emmagatzematge de dades, en una seqüència de dimensions fixes d’un mateix tipus d’element.
- “Blockchain”: Sistema que registre transaccions realitzades amb criptomonedes en varis ordinadors vinculats mitjançant una xarxa P2P.(2)
- “Bitcoin”: Moneda digital que registra transaccions i genera noves unitats d’aquesta mitjançant la solució computacional de problemes criptogràfics, operant de forma independent a un banc central. (3)
- “BitTorrent”: Protocol de transferència d’arxius P2P per compartir dades a través d’internet, en que es transfereixen les parts descarregades entre usuaris.
- “Bytes”: Grup de dígit binaris o bits, operats com una unitat o element.
- Consorci: Associació d’empreses o entitats amb els mateixos fins, per participar en un projecte o negoci de forma conjunta.(4)
- “Constructor”: Funció especial inicialitza variables d’estat d’un contracte.
- Cooperativa: Societat formada per consumidors, productors i venedors, amb la finalitat de produir i comerciar d’un mode més favorable per tots.
- Dapp o aplicació descentrelitzada: aplicació informàtica que aporta una funció o utilitat a l’usuari, que funciona de forma autònoma mitjançant l’ús de contractes intel·ligents executats dins una “blockchain”
- “Deploy”: nomenclatura informàtica per referir-se a la implementació d’un contracte intel·ligent a la “blockchain” corresponent mitjançant una transacció i el codi compilat.
- “External”: Tipus de funció que permet la interacció externa al contracte amb ella.
- Fiat o diner fiduciari: moneda emesa pel govern i suportada pel govern corresponent.
- Figura jurídica: Modalitat legal d’un concepte o entitat.
- “For”: element que repeteix un bloc de codi especificat durant un nombre específic de vegades.
- “Hash”: Resum d’una entrada o missatge produït a través d’un algoritme matemàtic unidireccional, la qual es pràcticament inviable invertir el càlcul.
- “If”: Element de control fonamental que permet realitzar decisions i executar-les condicionalment.

- “KeyType”: Valor clau a utilitzar per obtenir l’estructura de dades corresponent dins un “mapping”.
- “Mapping”: Element que realitza la funció com de diccionari, emmagatzemant parells de clau-valor.
- “Memory”: Paraula clau que especifica la creació d’una porció d’espai per aquella variable, garantint les seves dimensions i estructures.
- “Modifier”: Element que conte codi font del contracte que pot modificar el comportament de funcions a les quals s’adjunta.
- “Msg.sender”: Adreça o “address” de la persona que estigui connectada actualment amb el contracte.
- “Optimal Power Flow”: Model matemàtic que representa un problema de determinació dels nivells d’operació d’una central elèctrica, per tal de minimitzar els costos d’operació.
- “Payable”: “modifier” o modificador, que permet a la funció realitzar transaccions amb valors diferents a 0 ETH.
- “Private”: Element que nega l’accés a d’altres contractes intel·ligents a les variables, funcions o elements adjunts.
- “Peer-to-Peer”: Tipus de xarxes on tots els membres operen i es relacionen d’igual a igual.
- “Proof-of-Stake”: Protocol o mecanisme de consens per a “blockchain” on es seleccionen validadors en proporció a la quantitat de criptomoneda associada, evitant així el cost computacional del “Proof-of-Work”.
- “Proof-of-Work”: Prova d’esforç computacional en forma de prova criptogràfica on es demostra que s’ha gastat una quantitat específica de poder computacional per resoldre el problema.
- “Public”: Element que permet a tothom, ja sigui des d’un contracte o persona externa, accedir a la variable, funció o element adjunt.
- “Return”: Paraula clau que retorna els elements especificat en ell i del mateix tipus i quantitat que els especificats en el capçal de la funció.
- “Returns”: Element on s’especifica la quantitat i tipus de variables a retornar abans del bloc de codi corresponent.
- “Scripts”: Terme informal per descriure un programa senzill.
- “Smart-contracts” o contractes intel·ligents: Programa informàtic o protocol de transaccions que executa, controla i registra de forma automàtica esdeveniments i accions rellevants pels termes de l’acord d’un contracte.
- Societat Mercantil: Entitat constituïda per una o varies persones sota una denominació o raó social, que aporten capital i buscant un mateix fi lucratiu.
- “String” Element que emmagatzema un tipus de dades que pot contenir des de números i espais a lletres.

-
- “Struct”: Element que permet crear diferents tipus de variables, sent aquestes un conjunt d’altres variables o elements que emmagatzemen dades o valors en forma d’estructura.
 - “Urban Innovative Actions”: Iniciativa europea que promou projectes pilot per al desenvolupament urbà sostenible.
 - “ValueType”: Valor vinculat a una clau a utilitzar per obtenir l’estructura de dades corresponent dins un “mapping”.
 - “View”: Modificador que assegura que la funció adjunta no modifica l’estat de la “blockchain”.
 - “Wallet” o “Crypto Wallet”: Cartera on emmagatzemar les claus privades de forma segura, permeten enviar rebre i gastar criptodivises.
 - “Whitepaper”: Document tècnic que conté l’informació principal d’un projecte de tecnologia “blockchain”, com el seu objectiu i la tecnologia a utilitzar per aconseguir-lo.

Acrònims

- AMB: Àrea Metropolitana de Barcelona
- BTC: Moneda digital de la “blockchain” “Bitcoin”.
- CEL: Comunitat Energètica Local
- CER: Comunitat d’Energies Renovables
- CCE: Comunitat Ciutadana d’Energia
- DERs: “Distributed Energy Resources” o Recursos Energètics Distribuïts
- DR: “Demand Response” o sota demanda
- FEDER: Fons Europeu de Desenvolupament Regional
- FT: “Fungible Token”
- MCP: “Market Clearing Price” o preu de cassació de mercat
- NFT: “Non-Fungible Token”
- OPF: “Optimal Power Flow”
- PIMES: Petites i Mitjanes Empreses
- PoS: “Proof-of-Stake”
- PoW: “Proof-of-Work”
- P2P: “Peer-to-Peer”
- TPS: Transaccions per segon
- UE: Unió Europea
- UIA: “Urban Innovative Actions”
- VPP: “Virtual Power Plant” o central elèctrica distribuïda virtual



Índex

RESUM	I
RESUMEN	II
ABSTRACT	III
GLOSSARI	V
Conceptes.....	v
Acrònims	vii
1. INTRODUCCIÓ	3
1.1. Objectius del treball.....	6
1.2. Abast del treball.....	6
2. ESTAT DE L'ART	7
2.1. Literatura científica	7
2.2. Absències i contribució	11
3. DESCRIPCIÓ I FUNCIONAMENT D'UNA COMUNITAT ENERGÈTICA LOCAL	13
3.1. C. E. L. Vilawatt.....	15
4. SISTEMA ELÈCTRIC I ENTORN LEGAL DE L'AUTOCONSUM COL·LECTIU	18
4.1. Llei 24/2013, del 26 de desembre, del Sector Elèctric.....	18
4.2. Reial Decret-Llei 15/2018, del 5 d'octubre	19
4.3. Reial Decret 244/2019, del 5 d'abril.....	21
4.4. Ordre TED/1247/2021 del 15 de novembre	23
5. TECNOLOGIA "BLOCKCHAIN"	25
5.1. Plataformes i els seus llenguatges de programació	29
5.2. Introducció a Solidty	31
6. DISENY DAPP	35
6.1. Registre usuaris.....	36
6.2. Alfa List	39
6.3. Facturació.....	48
7. SIMULACIÓ DE LA DAPP DISSENYADA	53
7.1. Escenari 1	54



7.2. Escenari 2.....	55
7.3. Escenari 3.....	57
7.4. Comparativa Resultats.....	62
ANÀLISI DE L'IMPACTE AMBIENTAL	65
CONCLUSIONS	67
Resultats i Contribucions.....	67
Limitacions i treball futur.....	68
Conclusió final.....	68
PRESSUPOST I/O ANÀLISI ECONÒMICA	71
BIBLIOGRAFIA	73
ANNEX A	79
A1. Codi Registre Usuaris.....	79
A2. Codi alfaList.....	81
A3. Codi Facturació.....	87
A4. Dades Escenari 1.....	91
A5. Dades Escenari 2.....	93
A6. Dades Escenari 3.....	97

1. Introducció

En els últims anys, el sector elèctric s'ha vist sotmès a forts canvis regulatius i tecnològics, donant lloc a una transició energètica global. Aquests canvis inicien a partir de la implementació de tecnologies renovables per a la producció d'energia. A partir d'aquest punt i amb el pas dels anys la consciència social sobre l'escalfament global i el canvi climàtic ha anat creixent de forma exponencial fins arribar a esdevenir un dels punts més importants en l'agenda de tots els governs a nivell mundial. Per tal de reduir l'impacte en els mercats energètics nacionals d'indústries productores d'energia amb altes emissions de CO₂ i gasos d'efecte hivernacle, reduint així les emissions nacionals, s'ha incentivat l'ús i la producció d'energia renovable, essent les més destacades la producció d'energia eòlica i la solar.



Il·lustració 1. Pacte climàtic del COP26 a Glasgow (Font: (5))

Conjuntament a l'escenari anterior, la consciència social eixida ha comprès l'abast del problema mundial que feia temps que els experts avisaven. Aquest fet ha potenciat enormement la implicació dels usuaris finals en el seu consum d'energia i en la seva manera de viure, buscant alternatives més sostenibles, per a la seva vida diària. Un altre efecte de la consciència social és la implicació política dels governs per la millora d'aquesta situació. Un exemple n'és la cimera climàtica del COP 26 a Glasgow el passat Novembre (Il·lustració 1), on es pot observar tots els representats polítics presents.

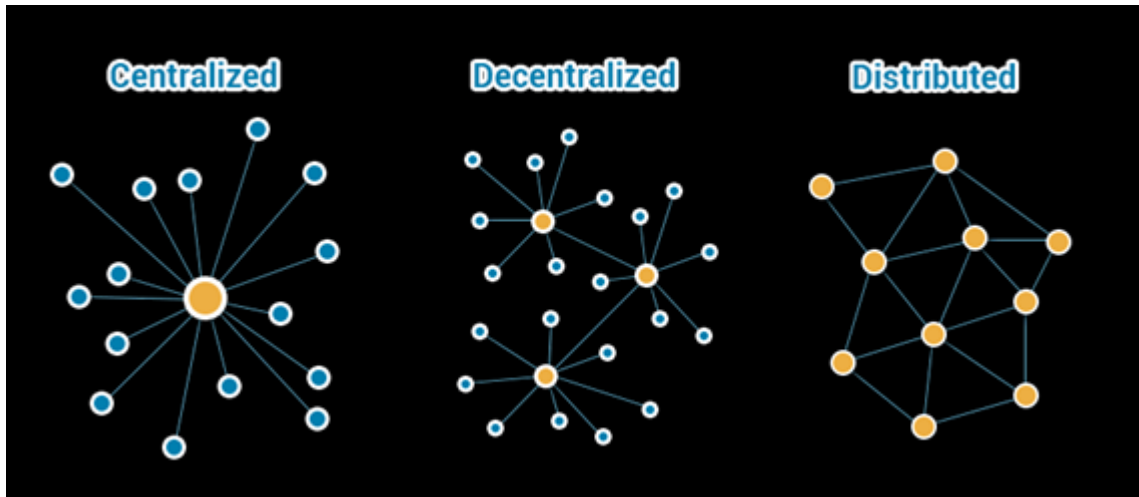
A tots els fets anteriors, és necessari afegir i puntualitzar que encara ha agafat més importància la comprensió de la ciutadania de l'abast del problema mundial en que ens trobem, en la situació de pandèmia actual. Aquesta, si que ha servit per inicialment reduir les emissions degut al confinament que han aplicat diversos països, però en el procés de la reactivació de l'economia, s'ha no ha fet més que accentuar la gravetat del canvi climàtic que vivim i la necessitat de posar-hi remei,. També, degut a la reactivació econòmica viscuda posterior a les primeres onades de la pandèmia, els preus de l'energia, vinculats al de matèries primeres com el gas, gasoil o al barril de cru, s'han incrementat de forma astronòmica fins el punt que actualment el valor de l'energia elèctrica es un 440% major respecte fa un any.

En el context de la transició energètica global, s'està canviant el mode de treballar dels sistemes energètics. Es passa de, actualment de forma generalitzada tenim una producció d'energia centralitzada, amb fluxos d'energia unidireccionals dirigits cap als consumidors passius, cap a un sistema elèctric distribuït i descentralitzat. L'objectiu d'aquest canvi és arribar a una generació i emmagatzematge distribuïts i descentralitzats, i amb un sistema capaç de transmetre energia de forma bidireccional.

Amb la gran incorporació d'energies renovables dins el mix de producció energètic, i el creixent nombre de productors locals d'energia solar, es crea una base on sustentar-se una generació distribuïda i descentralitzada. Per fer-ho possible, és necessària una nova arquitectura de control Per això es t'he en el punt de mira el comerç flexible a mode local a la xarxa per tal d'optimitzar relació entre l'oferta i la demanda.

Paral·lelament, en la branca de la informàtica del sistemes descentralitzats ha nascut l'anomenada tecnologia "*blockchain*", amb l'objectiu de transformar el món de les transaccions digitals. Aquesta tecnologia ha despertat un gran interès en la població que fins a temps prepandèma no en tenien coneixement, vivint un creixement exponencial molt important en els últims 3 anys.

Actualment la gran majoria de plataformes digitals estan estructurades de forma centralitzada, a través d'on s'accedeix a les diferents prestacions i/o serveis que ofereixen les diferents plataformes. Aquesta estructura es construeix a través de nexes centralitzats en servidors que gestionen i coordinen la major part del processament de dades de la xarxa de les diferents plataformes. Això permet una fàcil gestió d'aquesta tot i que té alguns inconvenients: com a exemple, quan es produeixen errors en algun dels nexes centrals, cau tota la plataforma que esta suportada per aquest. Un altre punt a tenir en compte és la disposició que tenen els usuaris, de gestionar, decidir i controlar la plataforma. En sistemes centralitzats, els usuaris no tenen tals capacitats i totes elles estan acumulades en una única entitat. En la Il·lustració 2, s'observa una representació gràfica d'un sistema centralitzat en un node central, un sistema descentralitzat en diversos nodes centrals i un sistema distribuït on tots els usuaris exerceixen de nodes centrals. (6)



Il·lustració 2. Visualització dels tipus d'arquitectura dels sistemes informàtics (Font: (7))

Totes les plataformes i xarxes digitals estan basades en la transferència de dades i informació de manera casi instantània a través d'intermediaris i servidors centrals on els usuaris dipositen la seva confiança. Això també permet la transferència d'actius digitals amb valor econòmic a través d'aquestes plataformes, però dependent d'una entitat externa i centralitzada per tal de realitzar-les, perdent el control sobre les transaccions durant cert període. El fet que l'única forma digital per tal de transferir valor sigui a través de tercers, es coneix com el problema *"Double spending"*. Aquest problema el resol Satoshi Nakamoto amb la creació d'un algorisme informàtic capaç de transferir actius digitals que no es poden copiar, al que avui ens referim com a tecnologia *"blockchain"* (8). (9)

La tecnologia *"blockchain"* es basa en una xarxa descentralitzada autogestionada, sense necessitat d'una figura central que ho coordini tot, de forma que tots els membres d'aquesta xarxa operen d'igual a igual o *"peer to peer"*. En ella es realitzen intercanvis d'actius econòmics entre persones, de mode que es crea una descentralització del sistema financer global. A través d'aquest algorisme neix *"Bitcoin"* com la primera criptodivisa.

L'intercanvi d'actius econòmics entre persones mitjançant criptodivises, és un dels diferents usos que té la tecnologia *"blockchain"*, pel qual es conegut. Altres aplicacions d'aquesta tecnologia van des de sistemes de votació per entitats esportives, ens municipals, creació de art digital, la gestió i governança d'organitzacions, intel·ligència artificial, arribant fins al sector energètic i la seva gestió.

1.1. Objectius del treball

Els objectius proposats en la realització d'aquest treball estan compresos en els següents punts:

- Entendre i exposar el funcionament d'una comunitat energètica, com una entitat generadora d'energia de forma descentralitzada i distribuïda, formada principalment per productors locals d'energia solar, i el marc legal que envolta aquesta entitat.
- Entaular les bases de l'ús de la tecnologia "*Blockchain*" com a eina per a la realització d'una relació flexible entre productors i consumidors de forma local, per tal d'optimitzar l'ús de l'energia i augmentar el benefici del territori.
- Creació d'una aplicació descentralitzada o "*Dapp*" des de zero, a partir d'un nivell de coneixement informàtic bàsic, per a la gestió, amb dades reals i instantànies, de la producció energètica d'una comunitat maximitzant el benefici tant econòmic com social aportat als usuaris d'aquesta.
 - Conèixer l'entorn que envolta les comunitats energètiques i les seves implicacions.
 - Dissenyar, desenvolupar i programar una base de "*smart contracts*" sobre la qual funcioni una aplicació descentralitzada.
 - Validar el funcionament d'aquesta mitjançant un petit estudi amb dades horàries d'un dia de 10 usuaris.

1.2. Abast del treball

Fent ús de la tecnologia "*blockchain*", basada en xarxes descentralitzades autogestionades i operades "peer to peer", es pretén dissenyar y estudiar la funcionalitat d'una Dapp dedicada al intercanvi comunitari i/o gestió d'energia dins la comunitat energètica.

L'abast del projecte va des del disseny, desenvolupament i programació de l'aplicació per a la gestió d'energia entre parts dins d'una comunitat energètica, la definició del marc legal que envolta aquesta i el terme de l'autoconsum col·lectiu aplicat a una comunitat energètica, i la demostració del seu funcionament ideat en el disseny mitjançant una petita simulació.

El treball no inclou el desenvolupament d'un algoritme de càlcul de preus, ni un model detallat dels elements físics que comporten i formen una comunitat energètica.

2. Estat de l'Art

En aquest apartat, es revisa i resumeix un recull de publicacions acadèmiques sobre les comunitats energètiques, mercats energètics i l'ús de la tecnologia "blockchain" en ells. Concretament, s'estudia la literatura sobre l'intercanvi d'energia, principalment de forma local, per tal d'autoconsumir-la i gestionar-la. A continuació es comenten les publicacions de forma breu per tal de conèixer el contingut que exposen.

2.1. Literatura científica

La literatura s'ha centrat en la gestió de l'energia i les diferents formes o mètodes per dur-la a terme. En alguns casos s'estudia més l'ús dels intercanvis "peer-to-peer" o s'implementen d'altres tipus de sistemes. Per exemple, en (10) es proposen opcions per a l'ús de núvols comunitaris per contribuir en sistemes d'eficiència energètica. Concretament en serveis basats en el núvol o "cloud-based" i la relació amb les xarxes intel·ligents o "Smart Grids". També es discuteixen i tracten les diferents iniciatives de recerca actuals, identificant els progressos més prometedors i dibuixar les possibles formes d'implementació. Seguint aquesta línia, en el cas de (11), es desenvolupa una plataforma mitjançant una planta d'energia virtual o "Virtual Power Plant" (VPP) basada en blockchain per facilitar transaccions energètiques entre usuaris residencials amb emmagatzematge, productors o amb càrrega variable. La simulació d'aquesta mostra una reducció del cost dels usuaris del 38,6% i del cost general del sistema d'un 11,2%. També en (12), un article de caràcter similar, es construeixen les bases d'un mercat d'energia "P2P". Aquest es realitza juntament amb sistemes bilaterals de subhasta ("k-DA") discriminatòria i uniforme, una comparativa d'aquests i les diferents estratègies de venda d'energia, simulat tot en un cas d'estudi amb 100 participants.

En alguns casos, l'escenari d'estudi es situa en microxarxes de corrent continu (CC). Com (13), on es defineix la relació entre microxarxa en CC i la tecnologia blockchain. Conjuntament es proposa un sistema amb potència dividida en bateries ió-liti i les carregues elèctriques a través de convertidors bidireccionals amb un controlador propi de lògica difusa, tot simulat a través de MATLAB Simulink.

D'altres autors, com en (14), simplement busquen les activitats més viables per a la implementació de la tecnologia "blockchain" dins el sector elèctric indonesi. Es determina que certs processos dins el sector es poden optimitzar i enriquir gracies a la implementació de la tecnologia "blockchain". Similar a l'anterior es troba (15), el qual analitza l'ús de tecnologia "blockchain" per reformar el sector elèctric xinès, característic pel monopoli del subministrament elèctric fins fa poc. S'estudia l'avanç d'aquest tipus de relació i s'investiguen casos de renom en el món energètic i l'ús de "blockchain", conjuntament amb els seus avantatges i desavantatges. Els autors⁹ de (16), van més enllà i realitzen una classificació

de 131 iniciatives reals de l'aplicació de tecnologia "blockchain" en l'àmbit energètic mitjançant una comparació entre les aplicacions més destacades, les seves taxes d'error i el nivell de detall tècnic i teòric per al desenvolupament real.

Alguns autor però, s'han centrat més en el registre de dades, sent el cas de (17), que ofereix una aplicació mitjançant "blockchain" dissenyada per guardar, seguir i analitzar dades energètiques d'edificis i comunitats públiques. Aquesta funcionaria com una base de dades de petjades energètiques pública. Els resultats preliminars de l'estudi de camp realitzats en un grup d'edificis públics destaquen els beneficis del seu ús generalitzat.

Una part important dels estudis consultats, realitza l'estudi i la viabilitat del desenvolupament d'una plataforma descentralitzada o també d'una Dapp, sobre la "blockchain" d'"Ethereum", per a la gestió d'energia en temps real. En destaca (18), pensada per un mercat energètic, basat en una estratègia d'oferta amb aprenentatge adaptatiu amb mecanismes de compensació dinàmica. Similar a l'anterior, es troba (19), pensat per a una microxarxa. Aquest implementant un mecanisme de comerç bilateral i utilitzant dades reals d'una comunitat de prosumidors o "prosumer community" d'Amsterdam amb restriccions físiques simulada com un problema de flux òptim de potència o "Optimal Power Flow" OPF. Seguint en la mateixa línia, en (20), s'investiga l'eficient aplicació de la tecnologia blockchain pel comerç "P2P" d'energia, tenint en compte la seva escalabilitat, seguretat i descentralització, proposant una millora respecte models base augmentat l'escalabilitat sense comprometre la seguretat i l'escalabilitat. D'altres autors com (21), simplement fan us de la tecnologia "blockchain" per a la realització d'un model o algoritme de compravenda d'energia "P2P", amb doble subhasta iterativa, maximitzant els beneficis individuals sense tenir un gran coneixement del mercat en sí. S'utilitza un escenari 24/7 amb 20 consumidors i 10 productors solars amb simulacions reals de consum i producció.

L'article (22), fa un pas més enllà i descriu l'enfoc del concepte d'una organització basada en "blockchain" d'una comunitat energètica local de baixa tensió. L'objectiu es utilitzar els recursos compartits per maximitzar el nivell d'independència energètica, i no centrada en la compravenda d'energia. Es donen les primeres estimacions de beneficis i costos i una agenda de recerca més ampla. En la línia de l'article anterior, (23) demostra l'ús de la tecnologia "blockchain" per a la creació i funcionament de comunitats energètiques. Es realitza la creació d'un entorn comercial més segur mitjançant "smart contracts" entre els productors i consumidors i proposa un marc energètic basat en blockchain per ajudar en l'intercanvi en una "community of prosumers".

En (24), es desenvolupa un marc sota demanda (DR) cooperatiu i descentralitzat per gestionar la compravenda d'energia diària sense comunitats d'edificis intel·ligents i amb l'ús de energia renovables local. S'utilitza un algoritme basat en "smart contracts" per connectar els diferents participants i realitzar la supervisió autònoma i la facturació. En (25), s'exposa un sistema distribuït de gestió lateral

de la demanda en una microxarxa comunitària, amb la integració de comptadors intel·ligents i energies renovables. La proposta minimitza el cost per als particulars i també pel conjunt de la comunitat.

A diferència dels anteriors, en (26), es proposa una plataforma blockchain de doble capa, que guardi de forma privada dades comercials i a part informació de lliure accés pública. Aquesta està basada en l'optimització de la interoperabilitat entre cadenes i amb les característiques necessàries per al comerç d'energia.

En l'article (27), s'estudien 3 casos d'usos de "P2P energy trading" en plataformes blockchain privades, desenvolupant l'estructura "blockchain" en el marc "Hyperledger" i definint els "smart contracts" utilitzats en el software. Seguint en l'ús de la "blockchain" "Hyperledger", l'estudi (28) desenvolupa un sistema basat en blockchain unificat per a la compravenda d'actius energètics entre "prosumers", vehicles elèctrics, companyies elèctriques i proveïdors d'emmagatzematge. S'implementen 2 versions del sistema en "Hyperledger Fabric", una modelada amb NFT i l'altra amb FT, i es realitza una comparació, amb diferents casos d'ús i els avantatges i desavantatges d'aquestes. Mitjançant la tecnologia "blockchain" anterior, els autors de (29) proposen una plataforma de mercat energètic basada en blockchain per a comunitats residencials. Amb l'objectiu de reduir la seva demanda pic i la factura elèctrica, presentant ofertes per a DERs durant un període diari, i amb una doble subhasta es cessa el mercat i s'obté el MCP. El model aconsegueix reduir la demanda pic en un 46 %.

Alguns d'aquests articles desenvolupen l'aplicació de la tecnologia "blockchain" en la compravenda d'energia, mercats energètics de diferents dimensions. Exposant els diferents beneficis i limitacions que comporta el seu ús. La seva lectura ha exposat els àmbits de desenvolupament i aplicació, la forma d'aplicació tècnica d'aquesta i el seu funcionament tècnic. Dins aquests destaquen principalment l'ús de l'ecosistema "Ethereum", conjuntament amb *Remix* (30), com plataforma per al desenvolupament de "smart-contracts" i aplicacions descentralitzades per als diferents objectius proposats. Una altra plataforma destacada per al desenvolupament de la tecnologia "blockchain" en el sector industrial ha sigut "Hyperledger" utilitzada en diferents dels articles consultats. A continuació la Taula 1, mostra una classificació entre els diferents projectes que han fet ús de plataformes "blockchain" per al seu desenvolupament.

	Plataforma Blockchain i Dapp	Determinació Preu	Simulació/Estudi
(26)	Ethereum	Promig de l'oferta +alta consumidor i +baixa productor	Estudi amb una comunitat de 500 edificis i dades reals
(24)		Cost local invariable en el temps o 0	NodeJS & ZeroMQ
(21)		Doble subhasta Mín Preu majorista Màx Preu minorista	entorn de simulació basat en múltiples agents
(22)		Calculat a través de Proves de funcionament	Estudi de 4 cases amb control Raspberry Pi
(18)		AR-C strategy	Comparació i simulació de AR-C vs ZI-C, AA, PA
(25)		Preu ofert pels productors	Testejada a la Blockchain
(19)		50% Preu de mercat al moment	Estudi numèric amb 8 escenaris diferents
(23)		-	Peersim
(20)	Ethereum; Qtum; Skyledger	Calculat mitjançant Smart contracts	Model empíric amb recollida de dades
(29)	Hyperledger Fabric	Doble subhasta	Experiment al Kortright Center Microgrid
(28)		a determinar segons l'oferta	Hyperledger Caliper v0.4.2
(27)		-	Funcionament i aplicació Raspberry Pi

Taula 1. Classificació de les publicacions en funció de la plataforma, mètode de determinació de preu o la simulació i simulació realitzats (Font: Propia)

Com s'observa en la Taula 1 de classificació, els projectes desenvolupats sobre diferents "Blockchain", segons els objectius del treball o la idea a desenvolupar, utilitzen diferents estratègies de determinació de preu per a la compravenda d'energia. Aquest fet afecta directament en la viabilitat econòmica que puguin tenir i en el tipus de projecte que es planteja.

En altres treballs també es proposa la temàtica sobre la gestió i les estratègies per la direcció de comunitats, micro-xarxes o xarxes energètiques de diferents nivells, d'on també s'obtenen coneixements, tant amb l'ús de la tecnologia "blockchain" com sense ella, els quals han modificat i ajudat a definir l'idea inicial a desenvolupar en aquest projecte.

Cal destacar l'expansió de la tecnologia "blockchain", i també de propagació de les comunitats energètiques en el nostre entorn en els últims temps, els quals poden deixar obsolets treballs, projectes o articles actuals i fer-ne sorgir de nous, de més rellevància en l'àmbit que es tracta.

2.2. Absències i contribució

Dins la literatura científica realitzada, s'ha observat una falta de consideració envers l'entorn legal on es desenvolupen les diferents plataformes o aplicacions. Aquest entorn no està unificat de forma internacional, de forma que segons el país de realització de l'article difereix considerablement, el qual també està vinculat directament a les polítiques del govern d'ocasió. En aquest cas el marc legal Espanyol, no és mencionat en la lectura científica realitzada. Addicionalment, en alguns casos com aquest, el marc legal canvia considerablement amb el pas dels anys, condicionant els projectes realitzats o a realitzar en un futur.

En aquest treball, es presenten 3 smart-contracts, desenvolupats mitjançant tecnologia "blockchain", concretament en la "blockchain" de "Ethereum". Aquests són una base per al desenvolupament de una Dapp per tal de que les comunitats energètiques emergents en el territori tinguin una eina fiable, automàtica, descentralitzada i distribuïda per a la seva gestió energètica i la conseqüent facturació de l'usuari. Les principals aportacions d'aquest treball són les següents.

- Aproxima l'usuari, o participant d'una comunitat energètica, al funcionament bàsic de l'aplicació descentralitzada mitjançant la que una comunitat energètica realitzarà la seva gestió i facturació.
- Exposar l'entorn legal Espanyol actual de l'autoconsum col·lectiu, i de les comunitats energètiques, i l'experimentació en l'ús de coeficients dinàmics *ex post*.
- Es desenvolupa el plantejament de l'escenari sobre el que treballar, el disseny de l'aplicació de gestió d'energia fent ús de coeficients dinàmics *ex post* basada en la "blockchain" d'"Ethereum" i es valida la funcionalitat del codi desenvolupat.

3. Descripció i funcionament d'una comunitat energètica local

El concepte de comunitat energètica local (CEL), és un terme que no inclou figura jurídica, i engloba els diferents conceptes de comunitats energètiques actuals. Aquests són una comunitat d'energies renovables (CER), i una comunitat ciutadana d'energia (CCE). Ambdós són figures jurídiques definides i establertes mitjançant directives europees.

Concretament es defineix el concepte de comunitat d'energies renovables en l'article 22 de la DIRECTIVA (UE) 2018/2001 DEL PARLAMENT EUROPEU I DEL CONSELL del 11 de desembre de 2018 (31). En el cas de les comunitats ciutadanes d'energia, s'estableixen mitjançant l'article 16 de la DIRECTIVA (UE) 2019/944 DEL PARLAMENT EUROPEU I DEL CONSELL del 5 de juny de 2019.

Les comunitats d'energies renovables (CER), són entitats jurídiques conformades mitjançant participació oberta i control efectiu dels socis/membres els quals seran persones físiques, petites i mitjanes empreses (PIMES) i/o autoritats locals de proximitat. El seu objectiu principal és l'obtenció de beneficis ambientals, socials i econòmics per als seus socis/membres i la zona d'operació d'aquesta. Tot això es duu a terme a través dels drets en producció, consum, emmagatzematge i venda d'energia produïda amb tecnologies renovables. També tenen dret a compartir l'energia produïda dins la comunitat mantenint els drets i obligacions dels membres d'aquesta com a consumidors. Finalment també tenen dret a accedir als mercats d'energia adequats de forma directa com mitjançant l'agregació de forma no discriminatòria.

Aquesta figura és introduïda en el marc normatiu espanyol, mitjançant el Reial Decret-Llei 23/2020, del 23 de juny (32), segons dicta la DIRECTIVA (UE) 2018/2001 DEL PARLAMENT EUROPEU I DEL CONSELL del 11 de desembre de 2018. Posteriorment es veurà la implicació d'aquesta legislació dins l'ordenament jurídic espanyol.

Les comunitats ciutadanes d'energia (CCE), són entitats jurídiques conformades mitjançant participació oberta i control efectiu dels socis/membres els quals seran persones físiques, autoritats locals PIMES. El seu objectiu principal és l'obtenció de beneficis ambientals, socials i econòmics per als seus socis/membres i la zona d'operació d'aquesta. Tot això es duu a terme a través dels drets en producció, consum, emmagatzematge i venda d'energia, sense limitacions en termes de la tecnologia de producció. També tenen dret a la prestació de serveis d'eficiència energètica, serveis de recàrrega de vehicles elèctrics o altres serveis d'eficiència energètica per als seus membres i socis.

Aquesta figura és introduïda en el marc normatiu espanyol, mitjançant el Reial Decret-Llei 23/2020, del 23 de juny, segons dicta la DIRECTIVA (UE) 2019/944 DEL PARLAMENT EUROPEU I DEL CONSELL del 5 de juny de 2019. Posteriorment es veurà la implicació d'aquesta legislació dins l'ordenament jurídic espanyol.

Les comunitats d'energia renovable i les comunitats ciutadanes d'energia, són dos figures molt similars, en termes d'organització i funcionament, com en els membres i persones que hi poden participar. També resideixen diferències entre elles, ja que les comunitats ciutadanes d'energia no estan limitades a energies renovables ni han de ser de caràcter local.

En termes d'entitats jurídiques, les comunitats energètiques locals, es poden constituir segons les seves necessitats i funcions en cooperatives, associacions o societats mercantils. Cadascuna d'aquestes entitats té les seves normes i característiques corresponents, les quals permeten certa flexibilitat en el model jurídic que seguirà la comunitat. D'aquesta forma es poden crear cooperatives amb finançament extern o amb finançament propi, associacions diverses amb la participació de l'ajuntament, o models mixtes entre els membres de la comunitat i l'ajuntament o una empresa privada, etc. En la Taula 2, s'exposen les normatives jurídiques referents per les entitats jurídiques esmentades, els articles més importants de caràcter local i la forma pròpia d'integrar un ens municipal dins la seva entitat, que segons les necessitats i objectius de la comunitat, poden facilitar-ne el procés.(33)

	Cooperativa	Asociación	Sociedad mercantil (SL)
Norma jurídica de referencia	Ley 12/2015 de cooperativas	Libro tercero del CCC	Ley de Sociedades de Capital
Normas relevantes por los entes locales	Art. 137.1 ROAS Art. 243.3 LMRLC Art. 86 y DA 9ª LBRL	Art. 72 LBRL Art. 47.2.c) Ley 40/2015	Art. 243.3 LMRLC Art. 137.1 ROAS Art. 86 y DA 9ª LBRL
Posibles pasos a seguir para articular la participación del Ayuntamiento	<ol style="list-style-type: none"> Para ejercer una actividad económica por medio de una cooperativa: - Art. 142 y ss del ROAS Para participar / adherirse como socio colaborador: - Acuerdo mayoría absoluta del pleno (art. 114.3.d) LMRLC) 	<ol style="list-style-type: none"> Para adherirse: acuerdo de la mayoría absoluta del pleno (art. 114.3.d) LMRLC) Posibilidad de firmar convenios de colaboración con la asociación y hacer aportaciones dinerarias o de terrenos (cesión de uso). 	Para ejercer una actividad económica por medio de una sociedad mercantil: - Art. 142 y ss del ROAS

Taula 2. Taula resum de possibles vehicles jurídics (Font: (34))

3.1. C. E. L. Vilawatt

Vilawatt és un consorci, estructura publico-privada, que neix a partir d'un projecte europeu de la UIA l'any 2016, que serveix per crear el consorci. Està format per dos administracions públiques, la AMB (15%), l'Ajuntament de Viladacans (65%), i 2 associacions privades (20%). Actualment ja funciona de forma 100% municipal (35).

Actualment les funcions de Vilawatt són:

Operador Energètic: organitzar compres agregades d'energia dirigides al sector residencial principalment, també al sector comerç i industrial, oferir servei de comercialitzador d'energia amb les comercialitzadores que guanyen les compres agregades, seguint criteris establerts com la producció 100% d'aquesta energia.

Oficina física com a agència d'assessorament, ofereix un servei d'assessorament i cultura energètica a la ciutadania, servei més ben valorat ja que resol dubtes sobre eficiència energètica o la instal·lació de panells solars, tenint aquest espai d'atenció. Des d'aquí s'engloba el projecte de comunitats energètiques locals.

Rehabilitació energètica d'edificis, amb la rehabilitació integral de 3 habitatges mitjançant criteris d'eficiència energètica, 3 proves pilots per tal d'observar si eren models explicables. Els resultats van ser molt positius a nivell d'estalvi energètic, i actualment s'està estudiant com oferir aquest servei a la ciutadania, i s'està vehiculant totes les consultes sobre l'eficiència a les llars oferint uns premis a la ciutadania i també mitjançant subvencions.

La Moneda Vilawatt és una moneda electrònica creada amb el projecte europeu per tal d'impulsar el comerç local, i capitalitzar els estalvis generats mitjançant la incorporació en les comunitats energètiques.

Des del consorci Vilawatt, s'està treballant en 3 projectes pilots de comunitats energètiques:

- Escola Enxaneta
- Mercat Plaça Europa
- Mancomunitat d'habitatges Can Preciós

A l'escola Enxaneta es crea la primera comunitat energètica al municipi per a compartir energia entre l'administració pública i la ciutadania, afavorint així el bescanvi d'energia entre el sector públic i el sector privat i també que sigui bidireccional. Això es realitza convertint una instal·lació existent de 100 kWp d'autoconsum individual en autoconsum compartit en xarxa pública, realitzat tots els tràmits i

modificacions tècniques corresponents. Es té el marc jurídic que engloba l'actuació redactat, i s'espera la publicació del concurs, de forma que es preveu l'adjudicació al primer trimestre del 2022.

Hi ha hagut dificultat en el marc legal perquè es comparteix un bé públic en el sector privat. Ja que l'energia que es genera en una instal·lació pública es considera de patrimoni públic, i al compartir-ho amb ciutadania i comerços ha d'estar molt ben regulat jurídicament per tal de complir totes les regulacions existents.

L'objectiu global del model es estendre'l a tota la ciutat a través de 23 instal·lacions fotovoltaïques, subvencionades pel fons FEDER que s'executaran al 2022, amb una potència total instal·lada de 1 MW. L'objectiu a curt mig termini, és que qualsevol ciutadà o ciutadana de Viladecans, que es vulgui sumar a les comunitats energètiques, tingui una instal·lació suficientment a prop, i es creu així ja que la disposició del nucli urbà es bastant concentrada, de forma que s'hi poguï connectar i beneficiar-se d'aquesta energia a través d'aquest model.

La instal·lació de l'escola Enxaneta, té una gran quantitat d'excedents degut principalment a la no coincidència entre producció i consum (estiu, caps de setmana, etc). Concretament es plantegen 3 fases pel projecte.

1. Creació d'una comunitat energètica a través d'una instal·lació d'autoconsum compartit per a que les comunitats de veïns/es properes a l'escola puguin aprofitar els excedents produïts per la instal·lació fotovoltaïca existent. S'està realitzant un concurs per trobar 50 famílies per aprofitar els excedents de l'escola.
2. Afegir a a la comunitat energètica la generació d'una instal·lació fotovoltaïca que s'executarà durant l'any 2022 a l'aparcament de Ponent, ubicat a una zona molt pròxima a l'escola.
3. A llarg termini s'estan creant mecanismes per ajudar a la ciutadania a crear instal·lacions fotovoltaïques d'autoconsum compartides en els seus habitatges o blocs de pisos. Afegir a la comunitat energètica la generació de 6 instal·lacions fotovoltaïques ubicades a les cobertes dels edificis de les comunitats de veïns/es. Integrant així la generació de les instal·lacions de les comunitats de veïns properes dins la comunitat energètica. Així ampliar el nombre de famílies al que es dona abast.

Actualment s'ha limitat a 50 el nombre de famílies per tal de que l'estalvi econòmic sigui significatiu mitjançant un pacte econòmic, per tal d'incitar la ciutadania a formar-ne part. S'han realitzat simulacions mitjançant el creuament de dades de generació i consum horàries mitjançant una eina pròpia.

En l'escenari 1 tenint únicament en compte la producció de l'escola i una mostra de consum de 50 habitatges. En total s'han generat 119.976 kWh/any, dels quals l'escola autoconsumeix un 56.944

kWh/any i els restants 62.852 kWh/any estan disponibles per als habitatges associats, corresponent un estalvi per habitatge de 49,963 €/any. S'aproxima un valor econòmic per energia de 0,03972 €/kWh.

Escenari 2, tenint en compte la generació de l'escola i de la futura pèrgola de l'aparcament i la mostra de consum de 50 habitatges. S'han generat 119.796 kWh/any corresponents a l'escola i 97.633 kWh/any corresponents a la pèrgola, dels quals l'escola autoconsumeix 62.950 kWh/any, i queden 154.479 kWh/any disponibles pels habitatges, creant un estalvi pels habitatges de 66,08 €/any. S'aproxima un valor econòmic per energia de 0,02139 €/kWh.

Escenari 3, tenint en compte la generació de l'escola, de la pèrgola, de les 6 instal·lacions comunitàries de 6 kWp en les cobertes de les comunitats de veïns i suposant una mostra de 50 famílies consumidores. Generant així 119.796 kWh/any per l'escola, 977.633 kWh/any per la pèrgola i 52.110 kWh/any a través de les instal·lacions comunitàries. De forma que l'escola autoconsumeix 64.476 kWh/any, i queden disponibles pels habitatges 205.073 kWh/any, corresponents a un estalvi del 70,96 €/any. S'aproxima un valor econòmic per energia de 0,01730 €/kWh.

Constitució de la comunitat energètica sobre l'entitat jurídica del consorci Vilawatt (C.E.L Vilawatt). L'ajuntament cedeix temporalment l'ús de la instal·lació i l'encàrrec de gestió al consorci Vilawatt, el qual anuncia unes bases reguladores, que incorporen els criteris socials que han de tenir els beneficiaris d'aquesta energia, per accedir a les quotes de participació de la instal·lació fotovoltaica. S'estableix una quota de participació del 0,03€/kWh aprox. amb la que s'han realitzat els càlculs de les simulacions, que ajuda a mantenir la viabilitat econòmica del consorci a llarg termini. Finalment la ciutadania per participar en les bases, s'associa al consorci Vilawatt a través de l'associació Ciutadana per a la transició Energètica.

4. Sistema elèctric i entorn legal de l'autoconsum col·lectiu

4.1. Llei 24/2013, del 26 de desembre, del Sector Elèctric

Segons l'article 1.2 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric (36), es defineixen el sistema elèctric com el conjunt d'activitats necessàries per al subministrament d'energia elèctrica juntament amb les persones físiques o jurídiques i societats mercantils que les duen a terme. Dins aquestes activitats, s'inclou la generació, el transport, la distribució, els serveis de recarrega energètica, la comercialització i l'intercanvi intracomunitari i internacional d'energia, i la gestió econòmica i tècnica del sistema elèctric. En l'article 6 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric, trobem definits els individus i societats que podran desenvolupar les activitats esmentades amb anterioritat.

- Els productors d'energia elèctrica, són les persones físiques o jurídiques encarregades de la generació d'energia elèctrica i de la construcció, operació i manteniment de les instal·lacions de generació.
- L'operador del mercat, és la societat mercantil designada per a les funcions establertes en l'article 29 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric, segons el qual duu a terme la gestió del mercat diari i interdiari d'energia elèctrica.
- L'operador del sistema elèctric, és aquella societat mercantil designada per a les funcions establertes en l'article 30 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric, on es concreta que la seva funció és la de garantir la seguretat i continuïtat del subministrament d'energia elèctrica, així com la coordinació entre el sistema de producció i el sistema de transport.
- El transportista, essent la societat mercantil encarregada del transport d'energia elèctrica, de la construcció, manteniment i gestió de les instal·lacions de transport i altres funcions definides en l'article 36 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric.
- Els distribuïdors, que poden ser societats mercantils, cooperatives de consumidors o usuaris del sistema elèctric, destinat a la distribució d'energia elèctrica, de la construcció, manteniment i gestió de les instal·lacions de distribució assignades a establir l'energia elèctrica en els punts de consum i altres funcions definides en l'article 40 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric.
- Les comercialitzadores, que poden estar compreses per societats mercantils, cooperatives de consumidors o usuaris del sistema elèctric, i, tenint accés a la xarxa de distribució i transport d'energia elèctrica, compren energia per vendre-la als consumidors, o altres membres del sistema, o per a la realització d'intercanvis internacionals.

-
- Els consumidors, són persones físiques o jurídiques que adquireixen energia elèctrica per al seu propi consum, principalment a les comercialitzadores, tot i que també hi ha aquells qui directament compren l'energia elèctrica en el mercat de producció, anomenant-los així Consumidors Directes en Mercat.
 - Els gestors de càrregues del sistema, són societats mercantils que també esdevenen consumidors, els quals estan habilitats per a la revenda d'energia elèctrica per a serveis de recàrrega energètica.

Dins el sistema elèctric concretat prèviament, es defineix l'autoconsum d'energia elèctrica en l'article 9 de la Llei 24/2013, del 26 de desembre, del Sector Elèctric, com el consum d'energia elèctrica provinent d'instal·lacions de generació connectades a l'interior de la xarxa del consumidor o a través d'una línia directa d'energia elèctrica associada a aquest.

Dins aquest mateix article, es contempen diferents modalitats d'autoconsum

- **Subministrament amb autoconsum.** Modalitat aplicable quan un consumidor disposa d'una instal·lació pròpia de generació per al seu consum, connectada internament a la seva xarxa de subministrament, sense que estigui donada d'alta com una instal·lació productora. Es tracta així únicament de la figura de consumidor.
- **Producció amb autoconsum.** Modalitat aplicable en consumidors associats a instal·lacions de generació inscrites en el registre administratiu d'instal·lacions de producció d'energia elèctrica que esta connectada internament a la seva xarxa de subministrament. Es tracta així de la figura del consumidor i la del productor
- **Producció amb autoconsum a través de línia directa.** Modalitat aplicable al consumidor associat a una instal·lació de generació d'energia elèctrica que estigui connectada mitjançant un línia directa. Es tracta així de la figura de consumidor i la de productor.

També s'estableix, que en casos on la instal·lació estigui connectada al sistema elèctric, els seus titulars hauran de complir amb les obligacions corresponents com a consumidors i/o productors, i també se'ls hi atribuiran els càrrecs per terme de peatges d'accés a la xarxa, cost del sistema i costos pels serveis de suport del sistema per l'energia autoconsumida. Finalment es dictamina que tots els consumidors aollits a modalitats d'autoconsum s'hauran de d'inscriure en el registre administratiu d'autoconsum d'energia elèctrica.

4.2. Reial Decret-Llei 15/2018, del 5 d'octubre

Degut al canvi en el govern Espanyol, resultat de la investidura com a president del govern de Pedro Sánchez, el govern entrant inicia el procés cap a la a transició energètica mitjançant l'aprovació del

Reial Decret-Llei 15/2018, del 5 d'octubre (37), de mesures urgents per a la transició energètica i la protecció dels consumidors. Dintre aquest Reial Decret-Llei, es pretén eliminar les barreres regulatòries que desincentiven la implementació de l'autoconsum, i buscant potenciar l'ús de l'autoconsum renovable per reduir la factura energètica dels consumidors que n'instal·lin, i consegüentment reduir així la demanda del mercat majorista.

Per tal d'aconseguir els objectius proposats, en acord amb l'article 18 del Reial Decret-Llei 15/2018, del 5 d'octubre, es modifica la llei 24/2013, del 26 de desembre, del Sector Elèctric, principalment l'article 9 d'aquesta.

Seguint la modificació establerta per l'article 9 de la llei 24/2013, del 26 de desembre, del Sector Elèctric, la definició d'autoconsum, establerta en el punt 1, esdevé el consum d'un o varis consumidor d'energia elèctrica provinent d'instal·lacions properes als consumidors i associats a aquests. A continuació, en el mateix punt, es determinen que les modalitats d'autoconsum ja no seran les definides anteriorment, sinó que s'estableixen les següents:

- **Autoconsum sense excedents.** Quan la injecció d'energia a la xarxa de transport i distribució no sigui possible. En aquest cas únicament existirà la figura del consumidor.
- **Autoconsum amb excedents.** Quan apart de subministrar energia per l'autoconsum es puguin injectar excedents a la xarxa de transport i distribució. En aquest cas existirà la figura de productor i consumidor.

També, en el punt 2 establert per la modificació, es crea el concepte d'instal·lacions pròximes per a l'autoconsum, definides com a instal·lacions generadores connectades a la xarxa interna dels consumidors associats, connectades mitjançant una línia directa o connectades a la xarxa de baixa tensió provinent del mateix centre de transformació.

Un dels punts d'estacats d'aquesta modificació, es situa en el punt 5, on s'exempten tots els càrrecs i peatges a l'energia d'origen renovable, cogeneració o provinent de residus. Però es determinarà una quantia si es realitza algun ús de la xarxa de distribució degut a instal·lacions pròximes per autoconsum. Complementàriament, s'exempta del permís de connexió i accés per a les instal·lacions d'autoconsum sense excedents on l'usuari associat ja en tingui per al seu propi consum. Finalment es preveu el desenvolupament reglamentari per tal d'obtenir mecanismes de compensació entre els dèficits d'energia dels autoconsumidors i els excedents d'aquests.

4.3. Reial Decret 244/2019, del 5 d'abril

Posteriorment a la publicació del Reial Decret-Llei 15/2018, del 5 d'octubre on es duu a terme un gran canvi en la regulació de l'autoconsum, es publica el Reial Decret 244/2019, del 5 d'abril (38). En ell es regulen les condicions administratives tècniques i econòmiques de l'autoconsum elèctric, es defineixen conceptes anteriorment utilitzats com el d'instal·lacions pròximes a efectes d'autoconsum, i d'altres de nou ús com el d'autoconsum col·lectiu. També s'especifiquen els mecanismes de compensació esmentats anteriorment en la modificació de l'article 9.5 realitzada en el del Reial Decret-Llei 15/2018, del 5 d'octubre, així com l'organització de la inscripció i coordinació de les dades del registre administratiu d'autoconsum elèctric.

Concretament, es defineix l'autoconsum col·lectiu com l'associació d'un grup de diferents consumidors, que de forma prèvia, han acordat alimentar-se de l'energia produïda per una instal·lació de producció pròxima a aquests i associada als mateixos. El model d'autoconsum col·lectiu és una classificació addicional a les modalitats d'autoconsum en funció de si es realitza de forma individual o conjuntament en un col·lectiu de consumidors. Tots els participants d'aquest hauran de pertànyer a la mateixa modalitat d'autoconsum i notificar-ho de forma individual a l'empresa distribuïdora, ja sigui a través de l'empresa comercialitzadora o directament, conjuntament amb l'acord firmat per els corresponents criteris de repartiment.

D'aquesta forma els consumidors participants del model d'autoconsum col·lectiu esdevindran consumidors associats, ja que estaran associats a instal·lacions pròximes de xarxa interior o a través de la xarxa de transport i distribució. Concretament, les instal·lacions de producció pròximes als consumidors i associades als mateixos, per acollir-se als models d'autoconsum vigents han de complir alguna de les següents condicions:

- Estar connectades als consumidors associats mitjançant la xarxa interior del propi consumidor o a través de línies directes.
- Estar connectades a qualsevol de les xarxes de baixa tensió derivades del mateix centre de transformació.
- La connexió dels consumidors i de la instal·lació generadora s'ha de realitzar en baixa tensió i a una distància menor de 500 m. Tal distància, esdevindrà la projecció ortogonal entre els aparells de mesura dels associats.
- La referència cadastral dels associats esdevingui la mateixa en els primers 14 dígits d'aquesta.

D'aquesta forma les instal·lacions pròximes i associades que compleixin la primera condició esdevindran instal·lacions pròximes de xarxa interior. I les instal·lacions pròximes i associades que compleixin alguna de les 3 últimes condicions es denominaran instal·lacions pròximes a través de xarxa.

Seguidament, s'estableix per aquest nou model d'autoconsum, el concepte d'energia horària autoconsumida individualitzada, fent referència a l'autoconsum net horari per un consumidor associat al model d'autoconsum col·lectiu o a una instal·lació pròxima. Aquesta, en el cas d'autoconsum col·lectiu, es calcularà de dues formes diferents.

- Si el valor de l'energia horària consumida individualitzada és major al valor de l'energia horària neta generada individualitzada, l'energia horària autoconsumida individualitzada obtindrà el valor d'aquesta última.
- Si és inferior, l'energia horària autoconsumida individualitzada obtindrà el valor de l'energia horària consumida individualitzada.

De la mateixa forma, s'especifica la definició d'energia horària consumida individualitzada, fent referència al consum total net horari del consumidor. Tal consum es mesura mitjançant comptador en la zona frontera de la instal·lació del consumidor.

Desglossant l'energia horària consumida individualitzada, es troba que és la suma de l'energia horària autoconsumida individualitzada i l'energia horària consumida de xarxa individualitzada. Aquesta última, es defineix com el saldo net horari d'energia que es rep des de la xarxa de transport i distribució. Tal energia tampoc procedeix ni d'instal·lacions pròximes ni d'instal·lacions associades al model d'autoconsum col·lectiu.

A continuació, es defineix l'energia horària excedentària individualitzada, esdevenint la diferència entre l'energia horària neta generada individualitzada i l'energia horària neta consumida per cada consumidor. L'energia horària neta generada individualitzada, és l'energia generada per una instal·lació associada a l'autoconsum col·lectiu o a pròxima a través de la xarxa que correspon al consumidor associat. Aquesta energia horària neta generada individualitzada es calcula de la següent forma:

$$ENG_{hi} = \beta_i \cdot ENG_h$$

- ❖ ENG_h correspon a l'energia neta horària total produïda per la instal·lació de producció associada.
- ❖ β_i es defineix com el coeficient de repartiment de l'energia generada entre els consumidors que participen en el model d'autoconsum col·lectiu. Aquest valor serà diferent per a cada usuari associat al model d'autoconsum col·lectiu, i serà el valor que figuri en un acord firmat per totes les parts associades i notificat a l'empresa distribuïdora. El mètode de càlcul d'aquest valor és indiferent, sempre que entre tots els participants associats i hi hagi un acord firmat. Tot i això, aquest valor ha de ser constant per a cada consumidor durant el mateix període de facturació, i la suma dels diferents valors d'aquest ha de ser sempre igual a 1. Aquest valor

posteriorment es el que aplicarà l'empresa distribuïdora com a encarregada de realitzar les lectures.

A partir d'aquí, i seguint la modificació realitzada per l'article 18 del Reial Decret-Llei 15/2018, del 5 d'octubre, en la Llei 24/2013, del 26 de desembre, del Sector Elèctric, en l'article 9 d'aquesta, s'estableix una modificació de la modalitat de subministrament amb autoconsum amb excedents. D'aquesta forma es divideix en la modalitat amb excedents acollida a compensació i la modalitat amb excedents no acollida a compensació. A la primera d'aquestes s'hi podran acollir aquells casos on el consumidor i productor optin per acollir-se a la modalitat de subministrament amb autoconsum amb excedents i compleixin totes les condicions recollides a continuació:

- La font d'energia ha de ser renovable.
- La potència de les instal·lacions de producció associades no pot ser superior a 100 kW.
- La realització d'un únic contracte de subministrament per al consum associat i el consum de serveis auxiliars.
- La realització d'un contracte de compensació d'excedents d'autoconsum.
- La instal·lació de producció no s'aculli a un règim retributiu adicional o específic.
- En el cas que no es compleixin totes les condicions esmentades, es pertanyerà a la modalitat amb excedents no acollida a compensació.

La realització del contracte de compensació d'excedents d'autoconsum vindrà vinculat a la comercialitzadora de l'usuari, de forma que es pot acordar entre les parts el preu de l'energia consumida i de l'energia horària excedentària. També en cap cas el valor econòmic de l'energia horària excedentària podrà ser superior al valor econòmic de l'energia horària consumida. I si els consumidors associats estan acollits a un mecanisme de compensació d'excedents, el productor no pot participar en cap altre mecanisme de venda d'energia. Cal destacar que els consumidors acollits al contracte de compensació d'excedents d'autoconsum, estaran exempts de pagar els peatges d'accés de l'energia horària excedentària individualitzada.

4.4. Ordre TED/1247/2021 del 15 de novembre

El passat 15 de novembre, es va aprovar l'Ordre TED/1247/2021 del 15 de novembre (39), on es modifica l'annex I del Reial Decret 244/2019, del 5 d'abril. Concretament, s'implementen els coeficients de repartiment variables en l'autoconsum col·lectiu. Això es deu a que en el Reial Decret 244/2019, del 5 d'abril, s'especificava que en tot cas els coeficients havien de ser constants per a totes les hores d'un període de facturació. Però en la disposició final cinquena d'aquest, es reserva el dret a modificar els annexos i el seu contingut. Seguint així l'esmentada cinquena disposició, es modifica l'annex I del Reial

Decret 244/2019, del 5 d'abril, de forma que es permeti la implementació dels coeficients de repartiment variables horaris (β_{hi}) per a l'autoconsum col·lectiu.

Aquests coeficients de repartiment variables són modificables al llarg de l'any i es poden classificar en 2 tipus.

- Coeficients de repartiment variables horaris *ex ante*, que són calculats de forma prèvia al període de consum.
- Coeficients de repartiment variables horaris *ex post*, que són calculats posteriorment al període de consum.

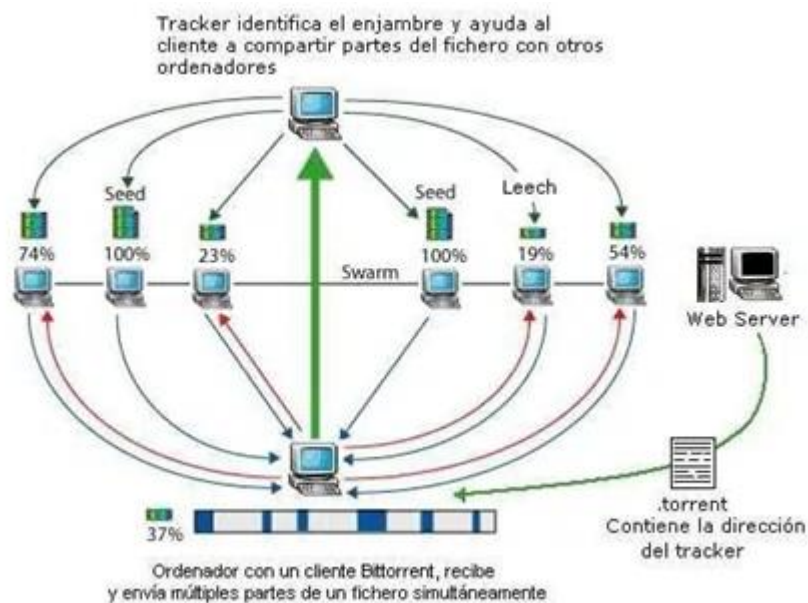
Actualment, l'Ordre TED/1247/2021 del 15 de novembre, només autoritza la implementació dels coeficients *ex ante*, ja que la implementació de coeficients dinàmics *ex post* es considera que aportarien més inconvenients que beneficis. Es justifica per la falta d'anàlisis, experiència i una major evolució de l'autoconsum col·lectiu.

5. Tecnologia “Blockchain”

Hi ha diferents efectes de l'ús d'una arquitectura centralitzada en serveis informàtics, com ara servidors o nodes centrals que serveixen per a la coordinació i gestió de les dades i el seu processat. Aquests provoquen que si es té algun tipus de problema o error en els nodes centrals cau la xarxa sencera. Un altre dels efectes de la centralització és que la capacitat de gestió i decisió recau en una única persona o un únic grup de persones com a autoritat màxima, amb objectius i opinions diferents a la seva comunitat de participants. Per evitar aquests problemes de la centralització, existeixen les xarxes descentralitzades.

Les xarxes descentralitzades es construeixen a partir d'una arquitectura descentralitzada, la qual utilitza la idea de que enlloc de tenir un únic node central, tots els usuaris d'aquesta actuen com a nodes, processant i gestionant les dades i de forma conjunta es coordinen per a la presa de decisions. D'aquesta forma si per qualsevol motiu un dels nodes participants caigués i deixés de funcionar correctament, la resta de la xarxa podria seguir treballant sense incidències. El fet que tots els membres formin part del col·lectiu que dirigeix la xarxa en dificulta la seva gestió i coordinació, ja que s'evita l'ús d'una figura central de confiança en la qual basar tots els processos de decisió i elecció, per exemple la recollida i recompte de vots, el cens electoral, etc. Per tal d'aconseguir un correcte funcionament del procés electoral que governi la xarxa, tot el col·lectiu que en formi part ha d'assegurar-ne la seva veracitat, evitant fins i tot actuacions o elements malintencionats. Seguint la idea anterior s'aconsegueix que la xarxa de persones i/o ordinadors s'autogestioni sola, arribant a consensos sense la figura central que ho coordini. Aquest tipus de xarxes on tots els membres operen i es relacionen d'igual a igual, es coneixen com “peer-to-peer”(P2P).(40)(41)

Aquest concepte és al darrere de xarxes com “BitTorrent”, un sistema d'intercanvi d'arxius P2P, essent actualment un dels indrets d'internet més populars per a la distribució de contingut multimèdia. El seu funcionament no es basa en un servidor central, sinó en la descarrega d'un arxiu .torrent, que indica on estan ubicades les diferents peces que formen l'arxiu en els ordinadors dels diferents usuaris d'aquesta xarxa. Seguint en aquesta línia també hi ha membres connectats al ordinador per descarregar-ne peces pels seus arxius. D'aquesta forma tots els seus usuaris treballen donant i rebent parts d'arxius, d'igual a igual o P2P). En la Il·lustració 3 s'observa l'entramat exposat de la xarxa BitTorrent.



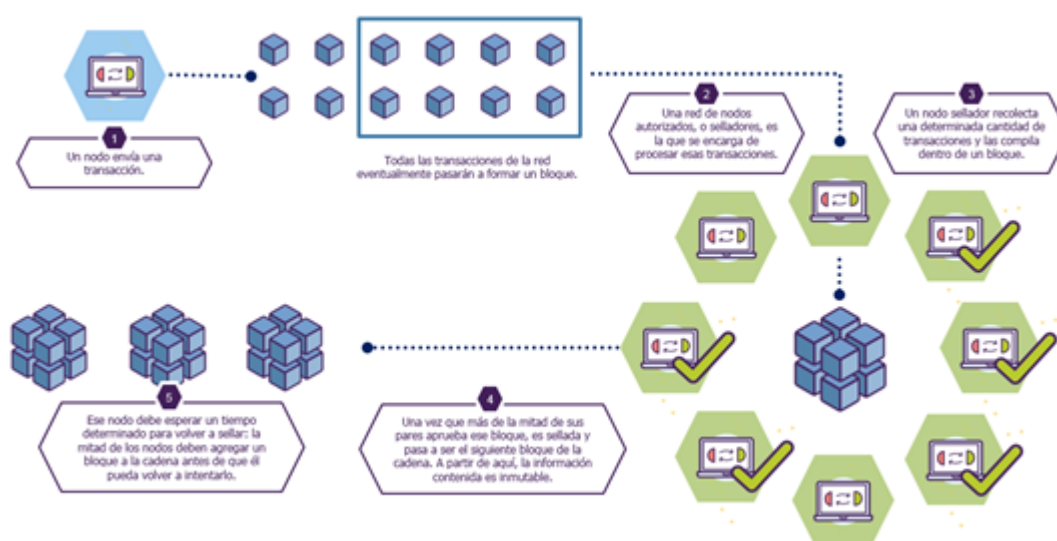
Il·lustració 3. Entremat de la xarxa BitTorrent (Font: (42))

Aquest concepte agafa popularitat a través de la publicació, el 2009, d'un Llibre Blanc o "*whitepaper*" anomenat "*A Peer-to-Peer Electronic Cash System*", publicat pel pseudònim Satoshi Nakamoto. Aquest proposa una solució algorítmica per a la creació d'una xarxa "*peer-to-peer*" per a l'intercanvi d'actius econòmics sense necessitat d'una institució central que els realitzi. Presentant així una proposta de descentralització del sistema financer a nivell global, del qual neix "*Bitcoin*". Al igual que a la xarxa "*BitTorrent*", on tothom emmagatzema parts d'arxius que es volien compartir, en "*Bitcoin*" es distribueix per tota la xarxa la informació corresponent als intercanvis econòmics realitzats entre els usuaris.

Quan ens referim a una criptomoneda com és "*Bitcoin*", el nom "*Bitcoin*" és el que li donem a tot el registre de transaccions o llibre de comptabilitat general. El funcionament d'aquest registre, es basa en que si l'usuari 1 (Joan) té 2 BTC(criptomoneda de l'ecosistema "*Bitcoin*"), i els vol transferir a l'usuari 2 (Marc), aquesta transacció quedarà guardada dins aquest llibre de comptabilitat general, essent aquesta informació distribuïda al llarg de la xarxa perquè tots els seus membres tinguin el llibre de comptabilitat general actualitzat. Ja que al mateix instant de temps que s'està realitzant aquesta transacció altres usuaris també n'estan realitzant des de diferents zones horàries, i sense dependre d'un servidor central per tal de coordinar l'hora de tota la xarxa, s'agrupen les transaccions durant un període de temps en blocs. Per tal de realitzar el que en un sistema centralitzat hagués de fer una entitat central com un banc, com verificar les transaccions, ara l'han de realitzar els diferents membres que conformen la xarxa. Un cop verificades les transaccions, s'agrupen en un bloc corresponent a un període de temps, i d'aquesta forma cada participant que estigui treballant en la xarxa haurà agrupat diferents transferències ja validades en un bloc. Així que s'obté un bloc diferent per cada participant

de la xarxa, el qual el conjunt d'aquesta haurà de decidir quin bloc incorpora al llibre de comptabilitat global. En aquest punt sorgeix un problema ja que és molt senzill fer passar 1 ordinador per centenars o milers d'ells, evitant així el correcte funcionament de la xarxa distribuïda, mitjançant la creació de direccions ip virtuals i amb objectius maliciosos, essent conegut això com un "*Sybil Attack*". Per solucionar aquest problema es fa pagar un cost computacional, a través d'un puzzle criptogràfic, per a cada membre que vol participar en la votació, d'aquesta forma si algú pretén realitzar un "*Sybil Attack*", haurà de pagar un cost computacional molt elevat per tenir èxit en el seu atac. Aquests mecanismes que t'obliguen a guanyar-te la teva participació en un procés electoral mitjançant la inversió de recursos computacionals, s'anomena com una prova d'esforç, de treball o "*proof-of-work*".

Aquesta prova d'esforç o "*proof-of-work*", es basa en la obtenció d'un Algoritme de "*Hash*" segur (SHA), el qual codifica qualsevol entrada de dades i/o arxius en una seqüència de números i lletres, concretament si s'utilitza el "*SHA-256*", s'obindrà un "*string*" o cadena de 256 bits. Aquesta cadena varia completament per a mínims canvis dins les dades o arxius d'entrada. En el "*proof-of-work*", s'utilitzarà el bloc d'agrupació de transaccions i la incorporació d'un enter numèric intern, per a la obtenció d'aquesta cadena "*Hash*". L'objectiu de la incorporació d'aquest enter numèric intern és que la seva variació faci canviar el "*Hash*" del bloc i així anar-lo canviant fins a resoldre el problema proposat per la prova de treball. Aquest objectiu proposat pel "*proof-of-work*" és el de l'obtenció d'un "*Hash*" amb una quantitat determinada de 0 inicials. Un cop aconseguida la combinació correcta, després de provar reiteradament i gastant molta computació, s'utilitza directament, sense procés de votació, el guanyador del "*proof-of-work*", el primer participant que obtingui el "*Hash*" amb la seqüència de 0 indicada, tindrà el dret de codificar el seu bloc amb el "*Hash*" obtingut i afegir-lo en el llibre de comptabilitat general, essent aquest el que posteriorment es compartirà amb la resta de membres de la xarxa. Ens referim a aquest fet com a que aquesta persona, el guanyador, ha minat un bloc. En el cas que es determini més d'un guanyador en el mateix instant de temps, es mantindran les dues versions del llibre general de comptabilitat fins que una d'aquestes versions aconsegueixi minar un nou bloc, evitant així el duplicat de transaccions, i així es determina que el llibre de comptabilitat general o la cadena de blocs més llarga és la correcta i la que s'utilitzarà. Donant així la recompensa pel treball realitzat únicament a l'usuari que ha minat el bloc que compon el llibre general de comptabilitat. En la Il·lustració 4, s'exemplifica el protocol de consens mitjançant "*Proof-of-Stake*" o PoS", el qual només poden participar una quantitat de nodes autoritzats i verificats. Al no realitzar una competició per minar un "*block*", es consumeix una menor quantitat d'energia. (43)



Il·lustració 4. Procés de consens mitjançant el "PoS" (Font: (43))

La recompensa als miners, usuaris que minen blocs, ve determinada per una transferència especial que afegeix el miner en el seu bloc com una recompensa per l'esforç realitzat. El conjunt de "*proof-of-work*" i incentius a través de criptodivises, permeten que la xarxa descentralitzada funcioni correctament i de forma segura.

La composició dels blocs comença amb el "*Hash*" de l'últim bloc minat, el conjunt de transaccions realitzades pels usuaris, la transacció especial de recompensa pel miner, l'enter numèric intern i finalment el "*Hash*" corresponent al bloc actual. D'aquesta forma, tots els blocs estan ordenats i encadenats un darrere l'altre creant així una cadena de blocs també anomenada "*Blockchain*". Essent la cadena de blocs o "*Blockchain*" una estructura de dades composta per blocs encadenats entre ells pel seu "*Hash*" i que degut al seu funcionament és immutable, ja que la variació de les dades ja registrades alteraria el "*Hash*" obtingut per el mateix bloc i per als següents.

Una aplicació de la tecnologia "*Blockchain*" és en la que es basa l'ecosistema de "*Bitcoin*" i la seva criptodivisa BTC. També ha guanyat popularitat l'ús d'aquesta tecnologia per a la creació de NFTs, altrament dit "*Non-Fungible Token*". Aquest es un mètode de certificació de propietat d'obres digitals o virtuals de manera que aquests arxius digitals es converteixen en actius transferibles a través de l'ecosistema corresponent. "*Ethereum*" és el principal ecosistema que permet desenvolupar aquest tipus d'activitat. Altres usos que té són els "*smart-contracts*" o contractes intel·ligents, que apart de registrar transaccions en els blocs minats, també s'hi adjunten peces de codi o "*scripts*" que s'executaran al complimentar-se un conjunt de condicions. Aquests contractes són molt similars als

contractes reals però sense intermediaris ni mediadors, ja que el contracte és públic i incorporat a la “*blockchain*”, el qual quan es compleixin les condicions sempre s’executarà.

5.1. Plataformes i els seus llenguatges de programació

Actualment dins l’àmbit de la tecnologia “*blockchain*”, hi ha diversos ecosistemes que permeten l’ús de “*smart-contracts*” o contractes intel·ligents dins la seva “*blockchain*”. Cadascun amb unes característiques, llenguatge, velocitat i permisos diferents. En la Il·lustració 5, s’adjunten els ecosistemes més destacats actualment i les principals característiques que els defineixen.

Top 6 smart contract platforms						
	Execution environment	Smart contract language	Turing completeness	Permission type	Consensus	TPS
Ethereum	EVM	Solidity	✓	Public	PoW [PoS expected]	~14–27
Polkadot	PEE	Depends on a selected chain	Depends on a selected chain	Depends on a selected chain	NPoS	~1,000 [100K with multithreading]
Hyperledger	Docker	Javascript, Go	✓	Private	CFT	~3,500
Tezos	Tezos VM	Michelson	✓	Public	PoS	~40
Stellar	Docker	NET, Scala, C++, Go	✗	Consortium	FBA [SCP]	~1,000
Solana	LLVM	C, Rust	✓	Public	PoH	~65,000 max

Il·lustració 5. Principals plataformes pel desenvolupament de “*smart-contracts*” (Font: (44))

Cada ecosistema té les seves pròpies característiques i diferències respecte la resta. Destaca l’ecosistema d’“*Ethereum*”, essent aquest el segon en capitalització de mercat darrera “*Bitcoin*”, i el qual s’ha utilitzat per al desenvolupament dels “*smart-contracts*” que formen la base de la Dapp dissenyada. Per entendre millor els ecosistemes esmentats en la taula a continuació se n’expliquen breument les seves característiques principals i els seus usos.

Comencem parlant de la plataforma “*Polkadot*”, la qual el seu principal objectiu és crear una xarxa d’interconnexions entre els diferents ecosistemes o plataformes “*blockchain*”. La seva “*blockchain*” està subdividida en diverses subcadenaes “*blockchain*” i subxarxes, amb les seves pròpies característiques com a ecosistema “*blockchain*” independent però alhora adjunt a la resta de subcadenaes i subxarxes. El nom d’aquestes subcadenaes és “*parachain*”. Llavors cada “*parachain*” té el seu propi llenguatge de programació del codi, té uns permisos diferents i una velocitat de procés de transaccions per segon (TPS) diferent, i és el que permet a la plataforma “*Polkadot*” desenvolupar

“*smart-contracts*”, ja que de forma nativa la seva “*blockchain*” no ho permet. A “*Polkadot*”, a diferència de “*Bitcoin*” o “*Ethereum*” que utilitzen com a protocol de consens entre els diferents membres el “*proof-of-work*” (PoW) comentat anteriorment, utilitza un protocol anomenat “*nominated proof-of-stake*”, el qual es basa en nomenar uns nomenadors i validadors dins el conjunt dels usuaris, els quals per desenvolupar aquesta funció se’ls hi bloquejarà els “*tokens*” o “*DOT*”, criptomoneda utilitzada a “*Polkadot*”, que tinguin en propietat, obtenint recompenses pel seu treball de forma prorratejada, o penalitzacions que reduirien la quantitat de “*tokens*” o “*DOT*” bloquejat i arribant a perdre aquesta garantia en cas de comportament negligent o advers. (45) (46)

Seguidament es tracta d’“*Hyperledger*”, una plataforma de codi obert, al igual que “*Ethereum*”, però orientada a desenvolupar tecnologia “*blockchain*” per a l’empresa privada. Tot i ser de codi obert alguns dels seus projectes més destacats estan liderats per IBM, Intel o Monax. Dins d’“*Hyperledger*” hi ha diferents entorns i eines per poder desenvolupar-se i cadascun ofereix un llenguatge i qualitats diferents. (47)

Seguint la Il·lustració 5, “*Tezos*” es una plataforma associada al “*token*” “*XTZ*” amb un protocol de consens similar al de “*Polkadot*” anomenat “*proof-of-stake*”, sense ús de mineria. Aquest permet crear “*smart-contracts*” i aplicacions descentralitzades en una “*blockchain*” d’una sola capa on combina les transaccions i els protocols de consens. Aquestes capes de les xarxes estan connectades a la “*blockchain*” i es comuniquen de forma ràpida i senzilla però de forma independent. D’aquest mode es permeten els canvis i modificacions en la gestió i actualització de la “*blockchain*” utilitzant llenguatge “*Michelson*”. (48)

“*Stellar*” és una plataforma descentralitzada de codi obert amb una moneda digital anomenada “*lumen (XLM)*” que permet transaccions entre qualsevol criptomoneda a fiat i viceversa, permetent de forma ràpida senzilla i segura l’intercanvi de qualsevol parell de criptomonedes. Al tractar-se d’una plataforma de codi obert, els seus projectes estan codificats amb diferents llenguatges, des de C++, a Python, Java o JS. El seu protocol de consens és propi i s’anomena “*Stellar Consensus Protocol*” (SCP) i està basat en un esquema tipus Acord Bizantí Federat (FBA). (49)

Al final de la Il·lustració 5 trobem “*Solana*”, una altra plataforma “*blockchain*” amb criptomoneda nativa anomenada SOL i que permet la programació de “*smart-contracts*” mitjançant els llenguatges C i Rust. Destaca per la seva gran escalabilitat de TPS i per fer ús d’un complement de mineria anomenat “*proof-of-history*” que encadena els “*Hash*” de les diferents transaccions un darrere l’altre que posteriorment s’incorporen dins un bloc, reduint substancialment l’esforç de minar un bloc. (50)

Finalment es parla d’“*Ethereum*” que és una plataforma global de codi obert, de forma que qualsevol pot participar en el desenvolupament del seu codi de forma gratuïta, per a aplicacions descentralitzades. La criptomoneda que utilitza aquesta xarxa és l’“*Ether*”, essent aquest un medi per

l'intercanvi de bens, serveis, i alhora esdevenir el llibre general de comptabilitat de la xarxa "Ethereum". Essent dels ecosistemes amb més experiència i altament utilitzats per al desenvolupament de Dapps, s'utilitza per desenvolupar els "smart-contracts" de base de la Dapp a dissenyar. El llenguatge en que es basa es el "Solidity", molt semblant a JavaScript i a Python, i ofereix eines i facilitats per a la creació de Dapps per a diferents aplicacions, des d'eines financeres, tecnologia, jocs o col·leccionables en forma d'art digital. El seu protocol de consens es basa en el "proof-of-work", tot i que degut a l'elevada quantitat d'energia que requereix aquest s'està estudiant i plantejant utilitzar el "proof-of-stake" i reduir així la petjada de carboni de l'ecosistema. (51)

5.2. Introducció a Solidity

Solidity és un llenguatge de programació estàtic, és a dir, que s'ha d'especificar el tipus de cada variable. Segons la ubicació de la variable, aquesta pot ser una variable d'estat, present en tot el contracte i definida al principi d'aquest, o local, present únicament dins l'agrupació de codi on estigui. Per exemple dins una funció, un "for" o un "if". Existeixen molts tipus de variables i també elements que permeten ajuntar diferents tipus de variables i crear una variable més complexa. També, es permet la interacció mútua entre diferents tipus de variables. (52)(53)

En Solidity no existeix el concepte de valor nul, tot i que les variables acabades de declarar obtenen un valor predeterminat segons el seu tipus.

A continuació s'exposen els diferents tipus de variables dins Solidity i els operadors lògics que accepta aquest tipus de variable.

Les variables tipus "bool" o booleans, únicament són valors constants "true" o cert i "false" o fals. Aquest tipus de variable pot interaccionar amb els operadors lògics de negació (!), conjunció o "i" (&&), disjunció o "o" (||), d'igualtat (==) i de desigualtat (!=). (2)(54)(55)

Les variables tipus "int" o "uint" són nombres enters definits o sense definir de diferents mides. Aquestes mides són múltiples de 8 i parteixen des de les variables tipus "int8" o "uint8" fins a la "int256" o "uint256". Cal esmentar que les paraules clau "int" o "uint" són àlies per a les variables d'aquests tipus de majors dimensions ("int256" o "uint256"). Accepten diferents operadors, com els que permeten realitzar comparacions entre variables (<=, <, ==, !=, >=, <) els quals retornen una variable booleana amb el resultat de la comparació, especificant si és cert o fals. També accepta operadors de bit i operadors de canvi de manera que $x << y = x * 2^{**}y$ o $x >> y = x / 2^{**}y$. També permeten la interacció d'operadors aritmètics per a la realització de sumes i restes (+ i -), només per a variables amb signe, i també per al càlcul de productes (*), divisions truncades sense decimals entre variables (/), la resta d'una divisió (%), o la potència de les variables i els exponents a definir (**). S'ha de destacar que per

a valors negatius únicament es pot treballar amb enters indefinits, sinó surt un error al compilar el codi. També per a les operacions aritmètiques, les mides dels enters resultants d'aquestes vindran dictaminades pel primer valor a definir o la base de la potència.

Actualment Solidity només permet la definició de nombres decimals definits o indefinits, altrament dit "fixed" o "ufixed". Les paraules clau "ufixedMxN" i "fixedMxN" permeten la declaració de variables amb una quantitat M de bits i N especifica la quantitat de decimals disponibles. Tot i això encara no es compatible per a l'assignació de valors a aquest tipus de variable.

Les variables tipus "address" o de direcció, permeten guardar en aquest tipus de variables adreces corresponents a la identificació pública dels usuaris d'"Ethereum". Dins aquest tipus de variable existeixen dos distincions, les que es defineixen com a "address payable", que permeten enviar-hi ETH i les que són "address" a on no es podran enviar ETH. Únicament accepten operadors comparatius (<=, <, ==, !=, >=, <).

En les variables "address", és possible l'aplicació de funcions i/o propietats per tal de que ens retornin informació referent d'aquesta. Aquest seria el cas de la paraula clau "balance" o balanç, que permet consultar el saldo d'una direcció d'"Ethereum" o la funció preestablerta "transfer", que permet el traspàs de ETH, en unitats de Wei, entre direccions. El Wei és la denominació més petita de l'ETH amb una relació de 1 ETH = 10¹⁸ Wei. En cas que el balanç d'ETH de la direcció des d'on es realitza la transacció fos menor a la quantitat a transferir o la direcció receptora rebutges la transferència, falla i es reverteix la transferència. Al revertir la transferència, es paga el cost pel gas utilitzat al intentar realitzar-la. El gas és el cost de recursos de càlcul de la "blockchain" d'"Ethereum".

També en referència a les variables tipus "address", es pot aplicar la paraula clau "send" o enviar, la qual realitza la mateixa funció que "transfer", però es un llenguatge dins el propi Solidity de més baix nivell o amb menys seguretat.

Per a la realització d'interaccions entre contractes existeixen funcions preestablertes com "call" o trucada, "delegatecall" o trucada delegada i "staticcall" o trucada estatica. Aquestes funcions permeten obtenir control sobre la codificació dels contractes a que es refereixen, i retornen un variable booleana amb la condició de l'èxit de l'operació i una variable de memòria tipus "bytes" que guarda totes les dades recollides de l'altre contracte. També es pot codificar informació o valors estructurats del contracte utilitzant les funcions "abi.encode", "abi.encodePacked", "abi.encodeWithSelector" i "abi.encodeWithSignature". Posteriorment es pot descodificar aquesta informació amb la funció "abi.decode" i especificant el tipus de variables en que es vol descodificar. (56) (57)

Els contractes, com a conjunt de codi, també es poden convertir en variables de tipus adreça i utilitzar aquesta "address" definida per interaccionar amb ell des d'altres contractes. D'aquesta manera només

s'interactua amb les funcions i variables d'estat que tinguin la seva visibilitat especificada en *"public"* o publica i *"external"* o externa.(58)

Dins el llenguatge Solidity, existeix un tipus de variable i referència que esta format per una matriu o seqüència de variables del mateix tipus. Aquest tipus de variable s'anomena *"array"* i quan es vol definir es posa 2 claus després de definir el tipus de variable de que estarà formada la matriu. En l'interior d'aquestes claus s'especifica la longitud d'aquesta matriu o seqüència (uint256[32]). Interactua amb operadors comparadors (<=, <, ==, !=, >=, >), operadors de bits i de desplaçament, permetent a aquests últims seleccionar una a una les variables dins l'*"array"* definit. També es permet no definir la longitud d'una variable d'aquest tipus i posteriorment amb el mètode *".length"* retorna el valor de la longitud de la variable especificada. També es poden realitzar matriu o *"arrays"* de pròpies matrius o *"arrays"* (uint256[32][32]). Tots els tipus de variables poden formar arrays, i fins i tot es poden fer dels elements *"mapping"* i *"struct"*.

Els *"string"* o cadenes són un tipus de variables que permet declarar com a variables cadenes d'escriptura utilitzant les cometes (') o doble cometes ("). Són convertibles a variables de tipus *"bytes"* si la longitud d'aquest ho permet.

Les *"function"* o funcions són el tipus de variable que s'assigna a les parts del codi que realitzen una funció concreta. Aquestes es divideixen en internes o *"intern"*, que només es poden utilitzar des del propi contracte, i externes o *"external"*, que es poden utilitzar des d'altres contractes. Aquestes paraules que denominen la visibilitat d'una funció podrien ser substituïdes per *"private"* o privat i *"public"* o públic respectivament. També s'especifica el tipus de funció segons la interacció que tindrà aquesta amb la *"blockchain"*, concretament s'ha despecificar si aquesta únicament llegeix informació de la *"blockchain"* (*"view"*), si interactua amb ella guardant-hi informació de variables (*"pure"*), o si s'ha de realitzar alguna transacció d'intercanvi d'ETH amb un usuari (*"payable"*).

Els elements que referencien altres variables estan formats per el *"mapping"* o mapeig, *"struct"* o estructura i l'*"array"* o cadena. Sempre que s'utilitza un d'aquests elements i es defineix el seu nom s'ha d'especificar en quin espai s'emmagatzemarà aquesta informació. Les àrees on es pot emmagatzemar aquesta informació són la memòria o *"memory"*, la qual esta limitada per la crida d'una funció externa, l'emmagatzematge o *"storage"*, on es guarden les variable d'estat del contracte i la seva vida útil està limitada per la del contracte, i *"calldata"*. Aquest últim és la ubicació on es guarden els arguments de la funció i cada funció té el seu propi *"calldata"*.

L'element de referència i variable tipus *"mapping"* o mapeig, es declara utilitzant el mètode *"mapping(_KeyType => _ValueType) _NomVariable"*. L'element *"_KeyType"* o clau pot esdevenir qualsevol tipus de variable a excepció d'elements de referència com *"mapping"*, *"struct"* o els *"array"*. El *"_ValueType"* pot definir-se com a qualsevol tipus d'elements, tan de referència com de variables.

La seva funció es pot imaginar com una taula amb tants eixos com claus es defineixin, de forma que s'assignen variables als diferents valors que obtingui la variable que realitza la clau. Seguint amb aquesta idea es pot realitzar un "mapping" amb un element de tipus variable que sigui un altre mapeig. D'aquesta forma s'obté un mapping amb 2 claus, de manera que cada configuració de claus se li pot assignar una variable. (58)(59)

Els elements de referència i variables tipus "struct" o estructura són un mètode proporcionat per Solidity per definir nous tipus d'elements mitjançant una estructura o conjunt d'elements base. Per cridar als elements dins d'una variable definida a partir d'una estructura, es cita el nom de la variable i a continuació després d'un punt el nom de l'element intern que es vol citar (NomVariable.NomElementIntern).

En Solidity també hi ha estructures de control que prenen decisions en funció d'una condició o més d'una, les quals són molt similars a la semàntica d'altres llenguatges com C o JavaScript. Un d'aquests elements és l'"if" o si. Concretament si la pregunta lògica suggerida és certa aplica la resta del codi contingut en ell. Sinó aquella part de codi queda anul·lada i prossegueix amb el codi següent. Un altre d'aquestes estructures de control és el "for", que permet el funcionament d'una part del codi durant el període que es compleixi una condició específica definida en el seu argument.

6. Diseny Dapp

Per al disseny de l'aplicació es planteja un escenari amb una comunitat energètica conformada per 10 usuaris consumidors, alguns també productors degut a instal·lacions solars en les seves cobertes. Aquests membres realitzen un acord firmat on acorden l'ús de coeficients de repartiment dinàmics *ex post*, utilitzats per calcular l'energia generada individualitzada de cada usuari. Aquest tipus de coeficients encara no estan vigents en la normativa actual, suposant així que la comunitat realitza també la funció de comercialitzadora dels seus membres.

Un cop concretat l'escenari, es plantegen 3 “*smart-contracts*” diferents.

1. Smart-Contract Realització del registre i dades bàsiques de l'usuari a la “*blockchain*” per al seu posterior ús en la tramitació del pagament de la factura i pagament pel servei realitzat.
2. Smart-Contract Recol·lecció de les dades necessàries, càlcul dels coeficients dinàmics i posterior registre d'aquests a la “*blockchain*”. En aquest “*smart-contract*”, es suposa que les dades utilitzades s'introduiran automàticament per “*smart-meters*” o comptadors intel·ligents.
3. Smart-Contract Importació de les dades obtingudes pels altres “*smart-contracts*”, càlcul dels diferents termes de la factura i realització del pagament d'aquesta amb l'autorització de l'usuari.

Seguint l'escenari plantejat, s'ha tingut en compte que per a la realització dels “*smart-contracts*” i els pagaments dels usuaris a la comunitat no es desenvoluparà en una moneda pròpia d'aquesta, com un “*token*”.

Es realitzarà mitjançant fiat, diner fiduciari, amb el que es suposarà una equivalència de $0.01\text{€} = 1000000 \text{ Wei}$, la denominació més petita de l'ETH amb una relació de $1 \text{ ETH} = 10^{18} \text{ Wei}$, per simplificar el funcionament. També s'establirà el preu de l'energia consumida de la xarxa al valor €/kWh proporcionat pel mercat majorista de l'electricitat. El cost del terme de potència per simplificar estarà establert en $0.1 \text{ €/kW}\cdot\text{dia}$, i es suposarà un impost elèctric del 10% conjuntament amb un IVA del 21%. El preu de de la compravenda dels excedents generats pels usuaris, s'establirà com el preu promig entre el preu del mercat majorista d'electricitat i el preu de venda d'excedents a la xarxa. Aquest últim concepte, en els casos on s'obtinguin ingressos per la venda d'aquests excedents, es tindrà en compte restant-lo de forma prèvia a la suma i aplicació de l'impost a l'electricitat i l'IVA.(60)

6.1. Registre usuaris

Inicialment es defineix la llicència, la versió de Solidity en la primera línia amb doble barra inclinada com a comentari. Es preferible la identificació SPDX de la llicència, que de caràcter estàndard s'utilitza la MIT. Seguidament es defineix el nom i el contingut del contracte el qual estarà limitat per la clau que hi haurà després del nom d'aquest i la clau de l'última línia del contracte.

A continuació declarem variables que serviran pel llarg del contracte, les quals es poden modificar o eliminar segons les necessitats de cada comunitat energètica, essent una d'aquestes tipus "address" amb nom "owner" el propietari del contracte, un tipus "uint256" amb nom "preu_servei" el cost de registrar-se en aquest contracte, i una variable anomenada "identificadors_contadors" també del tipus "uint256".

```
//SPDX-License-Identifier: MIT
pragma solidity >=0.5.0;
contract Registreusuaris {
    address owner;
    uint256 preu_servei;
    uint256 identificadors_contadors;
```

Il·lustració 6: Inici i declaració de variables d'estat del contracte Registreusuaris (Font: propia)

Seguidament es crea un "mapping", un mapeig o llistat de variables, que relaciona una variable tipus "uint256" que realitzarà la funció de clau, i un tipus de variable anomenada "SmartM" creada mitjançant l'element "struct". Dins aquest nou tipus de variable creat, hi constaran diferents dades, a les quals s'hi poden afegir d'altres si sorgís la necessitat, entre les que hi haurà l'identificador de l'usuari, el tipus d'instal·lació, o usuari de consum, el DNI de l'usuari, la "address" de l'usuari on es realitza la facturació, i els valors de kW de potència instal·lada i contractada en la seva tarifa de llum.

```
mapping(uint256 => SmartM ) public contadors;
struct SmartM {
    uint256 Identificador;
    string TipusI;
    uint256 DNI_usuari;
    address U_address;
    uint32 kWpi;
    uint32 kWpc;
}
```

Il·lustració 7: Declaració mapping contadors i struct SmartM del contracte Registreusuaris (Font: propia)

En les següents línies del contracte, es defineixen dos "modifier", condicions o requeriments que establirem a algunes funcions com que només les pugui realitzar el propietari del contracte (

anomenada “onlyOwner”), o que per realitzar aquella funció s’ha de pagar un preu (anomenada “preciofiltro”). Per realitzar això, es definirà un “require” dins a cada “modifier” on dins el parèntesi s’especifica una condició concreta per a cada cas, en un que el valor de la variable tipus “uint256” de nom “_precio” sigui igual al valor de la variable “preu_servei”. En el segon cas la condició serà que l’“address” del “msg.sender” sigui igual a l’“address” del propietari definida com a “owner”

```
modifier preciofiltro(uint256 _precio) {
    require(_precio == preu_servei);
    _;
}

modifier onlyOwner {
    require(msg.sender == owner);
    _;
}
```

II-lustració 8: Declaració modificadors del contracte Registreusuaris (Font: propia)

Un cop declarats els elements anteriors, realitzem el “constructor” del contracte, especificant dins aquest l’“address” corresponent a la variable “owner”, essent aquesta l’“address” que realitzi el “deploy” o activació a la “blockchain” del contracte juntament amb el valor del preu que es vol especificar que tingui aquest servei definit en la variable “_precio”. Seguidament tenim la primera funció del contracte, on inserim les dades de la variable tipus “struct”, mitjançant la creació interna de variables “uint256” i d’una variable interna “string” les quals cada una correspondrà a una dada dins la variable “SmartM”. Un cop escrites les dades per l’usuari i executada la funció aquesta, s’afegeixen les dades al llistat amb clau el valor e la variable “identificadors_contadors”. Finalment s’augmenta en una unitat la variable “identificadors_contadors”. Cal denotar que en aquesta funció es determina que és “public”, de forma que tothom pugui interaccionar amb ella, se li aplica el modificador “preciofiltro”, el qual l’usuari haurà de permetre el valor de tal transacció, i finalment la funció ha de ser “payable” per tal de permetre el pagament del seu servei.

```
constructor (uint256 _precio) {
    owner = msg.sender;
    preu_servei = _precio;
}

function addSmartM( uint256 _DNI_usuario, uint32 _kWpi, uint32 _kWpc, string memory _instalacio_tipus) public preciofiltro(msg.value) payable {
    contadors[identificadors_contadors] = SmartM(identificadors_contadors, _instalacio_tipus, _DNI_usuario, msg.sender, _kWpi, _kWpc);
    identificadors_contadors++;
}
```

II-lustració 9: Funció inicialitzadora i addSmartM del contracte Registreusuaris (Font: propia)

A continuació, creem una funció la qual ens permet observar, sense cap limitador ni cost, la quantitat d’usuaris que hi ha registrats en el servei. De forma que es pot saber la quantitat de claus dins el llistat definit anteriorment. La funció és “external” per tal que es pugui realitzar sense problemes, té la

funcionalitat “view”, ja que així ens permet llegir variables de la “blockchain” i retornar-nos el valor d’aquesta, definint el seu tipus dins el parèntesi del “returns”, en aquest cas un “uint256”.

```
function getidentificadors_contadors() external view returns(uint256) {
    return (identificadors_contadors);
}
```

Il·lustració 10: Funció `getidentificadors_contadors` del contracte `Registreusuaris` (Font: propia)

Finalment desenvolupem una funció que ens retorni els diferents valors de les variables numèriques introduïdes dins l’element tipus “struct” anomenat “SmartM”, retornant-nos així l’identificador utilitzat, el DNI de l’usuari en qüestió, la potència instal·lada en kWp i la potència de consum contractada en kW. Per realitzar l’anterior, determinem que la funció es “public” i “view”, però que només el propietari del contracte pugui accedir a tals dades amb el modificador “onlyowner”, això es pot modificar segons les especificacions de cada comunitat. A partir d’aquí especifiquem els tipus d’elements que volem que ens retorni al final de la funció, els quals seran 4 “arrays” d’un conjunt de valors tipus “uint256” i 1 “array” d’un conjunt de “address”, i utilitzem l’aplicació de la paraula clau “memory”, ja que quedaran guardats dins la memòria interna del contracte. A continuació creem els diferents “arrays”, els assignem un nom, i una longitud, determinada pel valor de la variable “identificadors_contadors”. Seguidament realitzem un bucle escanejant els diferents valors d’una variable interna “i” tipus “uint256”, fent que el màxim d’aquesta sigui el valor de la variable “identificadors_contadors”. Utilitzant aquest escaneig, es guarda dins als “arrays” definits prèviament els diferents valors de les variables que conformen l’“struct” guardat en el mapeig anomenat “contadors” que ens interessin. Un cop finalitzat el bucle i rastrejats tots els valors de la variable “i”, es retornaran els valors emmagatzemats dins els diferents “arrays” especificats entre els parèntesis de la funció “return.”

```
function getallSmartM() public view onlyOwner returns(uint256[] memory, uint256[] memory, address[] memory, uint256[] memory, uint256[] memory){
    uint256[] memory identificador = new uint256[](identificadors_contadors);
    uint256[] memory dni_usuari = new uint256[](identificadors_contadors);
    address[] memory u_address = new address[](identificadors_contadors);
    uint256[] memory kwpi = new uint256[](identificadors_contadors);
    uint256[] memory kwpc = new uint256[](identificadors_contadors);
    for (uint256 i = 0; i < identificadors_contadors; i++) {
        identificador[i] = contadors[i].Identificador;
        dni_usuari[i] = contadors[i].DNI_usuari;
        u_address[i] = contadors[i].U_address;
        kwpi[i] = contadors[i].kWpi;
        kwpc[i] = contadors[i].kWpc;
    }
    return ( identificador, dni_usuari, u_address, kwpi, kwpc);
}
```

Il·lustració 11: Funció `getallSmartM` del contracte `Registreusuaris` (Font: propia)

6.2. Alfa List

Inicialment es defineix la versió de Solidity en la primera línia. Seguidament es defineix el nom i el contingut del contracte el qual estarà limitat per la clau que hi haurà després del nom d'aquests i la clau de l'última línia del contracte.

```
1 pragma solidity ^0.7.6;
2
3 contract alfaList {
```

Il·lustració 12: Inici del contracte alfaList (Font: propia)

A continuació es defineixen amb nom i tipus les constants i/o variables que s'utilitzaran al llarg del contracte. Per tal de poder obtenir el valor dels coeficients dinàmics a descomptar en la factura dels usuaris, es defineixen 4 "mapping", cada un amb un nom diferent essent el primer Registre, el segon PreuMM, el tercer DadesH, i el quart DadesHI.

Els "mapping" o mapeig, és un element del llenguatge "Solidity" que relaciona i enllaça 2 o més elements. El primer element es defineix com "_KeyType". En introduir-lo, el mapeig retorna "_ValueType" amb el qual esta relacionat. Tot i així alguns tipus de variables no es poden assignar com "_KeyType" d'un mapeig, com per exemple un altre element tipus "mapping", un contracte, un "struct", "arrays" de mida dinàmica o un element tipus "enum". En canvi el "_ValueType" pot esdevenir qualsevol tipus d'element, inclòs un altre mapping.

En el nostre cas el "mapping" Registre, relacionem un enter indefinit tipus "uint256" amb un "Struct" anomenat BE, altrament dit balanç d'energia. Un "Struct" és un element que ens permet crear nous tipus d'elements finits, en forma d'una combinació de més d'un element bàsic de llenguatge com un enter indefinit "uint256", un adreça pública d'una "wallet" de "Ethereum", un "mapping", etc.

Tots els elements inicialment són 0, no tenen contingut o relacions assignades.

El "_KeyType" que utilitzarem en el "mapping" Registre correspondrà la quantitat de registres realitzats de forma prèvia al Registre actual, essent el "_KeyType" del Primer Registre 0.

L'element tipus "Struct" BE esta compost únicament per elements tipus enter indefinit "uint256", concretament 6 elements d'aquests tipus. Cada un d'aquests elements es correspondrà a una dada diferent.

El primer s'anomenarà "ID_user" i serà el DNI sense lletra de l'usuari al qui estigui vinculat el comptador intel·ligent d'on provenguin les dades.

El segon “uint256”, serà l’enter indefinit referent al codi de temps en el que s’ha produït el registre de les dades i tindrà per nom “Time”, agafant un valor entre 1-8760, corresponent a la quantitat d’hores que t’he un any natural.

El tercer enter indefinit ja es correspondria a l’energia produïda per l’usuari durant el període de temps definit pel codi temporal en Wh (EpWh).

Seguint en la línia, el quart valor esdevindria l’energia consumida per l’usuari durant el període de temps definit pel codi temporal en Wh (EcWh).

Com a penúltim element definim la relació entre l’energia produïda per l’usuari i l’energia consumida per l’usuari, en aquest període temporal definit en la variable “Time”, en tant per cent com a “Alfa”. D’aquesta manera existeix una “Alfa” per a cada registre de dades realitzat vinculada per el valor de la clau als diferents elements d’aquests “struct”.

Finalment, la diferència entre l’energia consumida i l’energia produïda es defineix com els “Excedents” que l’usuari ha produït durant aquest període temporal.

A continuació es defineix un “uint256” amb el nom de “addressRegistryCount” de manera que funcioni com a comptador de la quantitat de registres realitzats en el mapping.

```
mapping(uint256=> BE) public Registre;
struct BE {
    uint256 ID_user;
    uint256 Time;
    uint256 EpWh;
    uint256 EcWh;
    uint256 Alfa;
    uint256 Excedents;
}
uint256 addressRegistryCount;
```

Il·lustració 13: Declaració mapping Registre, struct BE i la variable addressRegistryCount del contracte alfaList (Font: propia)

Seguidament tenim la funció pública “set”, la qual rep uns valors externs, referents al ID_user, Time, EpWh i EcWh, i utilitzant com a “_KeyType” el valor de la variable “addressRegistryCount” (inicialment a 0). Es vincula amb l’element “struct” BE amb els valors externs rebuts per a les variables ID_user, Time, EpWh i EcWh, i utilitzant-les per calcular Alfa i Excedents. Per tal d’introduir elements externs al contracte, en aquests cas vindrien donats per comptadors intel·ligents, s’ha de definir el tipus de variable que està entrant i ser el mateix tipus de variable d’on s’ha de guardar. En el nostre cas tot són enters indefinits “uint256”.

```
function set(uint256 _ID_user, uint256 _Time, uint256 _EpWh, uint256 _EcWh) public {
    Registre[addressRegistryCount] = BE(_ID_user, _Time, _EpWh, _EcWh, 100*_EpWh/_EcWh, _EpWh - _EcWh);
    addressRegistryCount++;
}
```

Il·lustració 14: Funció set del contracte alfaList (Font: propia)

Seguidament es formula una funció per poder introduir els valors del preu de l'energia elèctrica segons el mercat majorista de l'electricitat, i vincular-lo al "mapping" PreuMM. En PreuMM es vincula el valor introduït amb una variable tipus "uint256" que correspondrà al període horari de cada valor.

```
mapping(uint256 => uint256) public PreuMM;
function setPreu(uint256 _temps, uint256 _PreuMM) public {
    PreuMM[_temps] = _PreuMM;
}
```

Il·lustració 15: Declaració mapping PreuMM i la funció setPreu del contracte alfaList (Font: propia)

A continuació es realitza una funció que té com a objectiu retornar tots els valors enllaçats i guardats amb els diferents valors de "_KeyType" del mapeig Registre. A l'hora d'obtenir els valors enllaçats en el mapeig, es pot introduir el nom del mapeig seguit de dos claus i a l'interior d'aquestes el "_KeyType" corresponent del qual es vol obtenir el seu "_ValueType". Aquesta acció únicament ens retornarà els valors registrats amb el "_KeyType" corresponent. Per tant es creen un conjunt d'arrays, amb una llargada igual a la variable "addressRegistryCount", per a cada element intern de l'element BE on si emmagatzemaran cada un dels valors proporcionats corresponents als diferents "_KeyType".

Per tal d'accedir a tots els "_KeyType" es realitza un "for" amb una nova variable interna i que anirà de 0 fins a l'enter més proper però no igual de la variable "addressRegistryCount" i cada vegada que realitzi el codi intern del "for" augmentarà la variable i en 1 unitat. Dins aquest "for" copiem en els arrays definits prèviament abans del for, els elements de BE per als "_KeyType" i de manera que quan el "for" passi per tots els valors de i, corresponents a tots els "_KeyType" existents, aquests es copien en els arrays.

Finalment ja fora del "for", però encara dins de la funció "getAll", utilitzem la sintaxi "return()" i a dins els diferents arrays definits a l'interior de la funció i separats per comes. Per tal d'utilitzar aquesta sintaxi, en la línia on es defineix el nom de la funció, s'ha de concretar que aquesta serà "public" de manera que de forma externa al propietari del contracte es pugui llegir. També s'hi ha d'incorporar la paraula "view" ja que el que estem realitzant en aquest cas és la lectura de valors escrits a la blockchain i exposant-los al exterior d'aquesta. Finalment s'ha de concretar els elements i el tipus d'aquests que es voldran retornar amb utilitzant la sintaxi "retorn()", en aquest cas a diferència de la part final de la funció té una s final.

```

function getAll() public view returns ( uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory ){
    uint256[] memory id_user = new uint256[](addressRegistryCount);
    uint256[] memory time = new uint256[](addressRegistryCount);
    uint256[] memory epWh = new uint256[](addressRegistryCount);
    uint256[] memory ecWh = new uint256[](addressRegistryCount);
    uint256[] memory alfa = new uint256[](addressRegistryCount);
    uint256[] memory excedents = new uint256[](addressRegistryCount);
    for (uint256 i = 0; i < addressRegistryCount; i++) {
        id_user[i] = Registre[i].ID_user;
        time[i] = Registre[i].Time;
        epWh[i] = Registre[i].EpWh;
        ecWh[i] = Registre[i].EcWh;
        alfa[i] = Registre[i].Alfa;
        excedents[i] = Registre[i].Excedents;
    }
    return ( id_user, time, epWh, ecWh, alfa);
}

```

Il·lustració 16: Funció getAll del contracte alfaList (Font: propia)

Seguint amb el contracte, a continuació definirem un mapeig anomenat DadesHoraries el qual utilitzant el codi de temps, de la variable Time, com a “_KeyType” s’enllaçarà a un conjunt de dades referents al conjunt dels usuaris durant el període T, referent al codi de temps utilitzant com a “_KeyType”, sota un element “Struct” amb nom DadesT, i seran les següents dades.

“amountET” =>Referent a la quantitat total d’excedents que tenen els diferents usuaris junts en el període T, definit com a un “uint256”

“amountUE” =>Referent a la quantitat total d’usuaris amb excedents, per tant Alfa ≥ 1 , en el període T, definit com a un “uint256”

“amountUA” => Referent a la quantitat total d’usuaris amb autoconsum sense excedents, per tant Alfa < 1 , en el període T, definit com a un “uint256”

“amountUC” => Referent a la quantitat total d’usuaris únicament consumidors, per tant Alfa = 0, en el període T, definit com a un “uint256”

“ENGh” => Fent referència a la quantitat total d’energia generada per la comunitat, durant el període h. És definit com un “uint256”.

“ENCh” => Fent referència a la totalitat dels consums de la comunitat durant el període h. És definit com un “uint256”.

“ENCPh” => Fent referència a la quantitat d’energia consumida que no s’autoconsumeix directament, dins el que es troba els consums dels UC i el dèficit energètic dels UA. És definit com un “uint256”.

```

mapping(uint256 => DadesT) public DadesH;
struct DadesT {
    uint256 amountET; // Quantitat d'excedents en l'hora h
    uint256 amountUE; // Quantitat d'usuaris amb excedents alfa >= 1
    uint256 amountUA; // Quantitat d'usuaris productors sense excedents 0 <= alfa < 1
    uint256 amountUC; // Quantitat d'usuaris únicament consumidors alfa = 0
    uint256 ENGH; // Quantitat d'energia generada en l'hora h
    uint256 ENCh; // Quantitat d'energia consumida en l'hora h
    uint256 ENCPH; // Quantitat d'energia consumida parcial en l'hora h =>
    // correspon a l'energia consumida dels UC més la diferència de l'energia consumida i l'energia produïda dels productors sense excedents
}

```

Il·lustració 17: Declaració mapping DadesH i struct DadesT del contracte alfaList (Font: propia)

A continuació creem una funció anomenada setDadesT, que ens servirà per relacionar el codi de temps utilitzat en el “_KeyType” amb el conjunt de l’element DadesT. Al realitzar càlculs a partir de valors introduïts en el mapeig Registre, la funció no té arguments. Per tant, en la primera línia de la funció no introduïm res dins els parèntesis d’aquesta i només posem que esdevindrà una funció pública, per tal de poder-la cridar des de altres smart-contracts de ser necessari. Finalment s’obren els claudàtors que defineixen el límit del codi dins la funció.

Seguidament utilitzem un “for” per a valors de t des de t = 0 fins que t és igual a l’últim valor del codi de temps(Time) en el mapeig Registre. Utilitzem “t++” per tal d’augmentar l’enter t en 1 unitat cada vegada que acaba el bucle amb el valor de t anterior.

Dins del “for” definim unes variables internes que seran les utilitzades posteriorment per indicar els valors de les variables de DadesT amb el codi de temps corresponent. A continuació i després de definir les variables, realitzem un altre “for” en aquest cas per a valors de i des de 0 fins a l’enter més proper però no igual de la variable “addressRegistryCount” i cada vegada que realitzi el codi intern del “for” augmentarà la variable i en 1 unitat.

Dins aquest nou “for”, introduïm un “if” que comprèn tot el codi intern d’aquest “for”. En ell es concreta que el codi següent només s’utilitzarà en cas que la variable interna t definida pel primer “for” sigui igual a la variable “Time” del mapeig Registre. Quan aquesta condició es compleixi, s’obren 3 “if” més, un per a cada tipus d’usuari, amb excedents, autoconsumidor i únicament consumidor. Segons la condició que compleixi el registre amb un “_KeyType” = i, sumarà els seus excedents a la quantitat total d’aquests i augmentarà en 1 unitat la quantitat d’usuaris amb excedents, augmentarà en 1 unitat la quantitat d’usuaris autoconsumidors i augmentarà el conjunt de consum parcial, definit anteriorment, en la quantitat corresponent a la diferència entre l’energia consumida per l’usuari i l’energia produïda per aquest, o augmentarà en 1 unitat la quantitat d’usuaris únicament consumidors i es sumarà el seu consum al conjunt del consum parcial. Finalment es sumaran tot el conjunt horari de d’energia produïda (_ENGh), i per l’altre costat el conjunt horari d’energia total consumida (_ENCh).

Un cop fora del segon “for” s’enllacen les definides en aquest amb el codi temporal(Time) t del primer “for”. D’aquest mode obtindrem tots els valors horaris d’aquests elements per als diferents períodes temporals.

```

function setDadesT() public {
    for (uint256 t = 0; t <= Registre[addressRegistryCount - 1].Time; t++) {

        // Declaració de variables internes per a la posterior introducció a DadesH

        uint256 _amountET;
        uint256 _amountUE;
        uint256 _amountUA;
        uint256 _amountUC;
        uint256 _ENGh;
        uint256 _ENCh;
        uint256 _ENCPh;

        for (uint i = 0; i < addressRegistryCount; i++){
            if ( t == Registre[i].Time){

                if (Registre[i].Alfa >= 100 ){
                    _amountET += Registre[i].Excedents;
                    _amountUE++;
                }
                if (0 < Registre[i].Alfa && Registre[i].Alfa < 100 ){
                    _amountUA++;
                    _ENCPh += Registre[i].EcWh - Registre[i].EpWh;
                }
                if (Registre[i].Alfa == 0 ){
                    _amountUC++;
                    _ENCPh += Registre[i].EcWh;
                }
                _ENGh += Registre[i].EpWh;
                _ENCh += Registre[i].EcWh;
            }
        }

        DadesH[t] = DadesT( _amountET, _amountUE, _amountUA, _amountUC, _ENGh, _ENCh, _ENCPh);
    }
}

```

Il·lustració 18: Funció setDadesT del contracte alfaList (Font: propia)

Finalment definim un mapeig anomenat DadesHI, en ell vincularem el “_KeyType” o clau a un altre mapping, de mode que tindrem un “mapping” amb 2 “_KeyType” on el primer serà “ID_user” de l’usuari en qüestió i el segon serà el codi de temps “Time” referent al període temporal per al valor que es definirà en el “_ValueType”. El “_ValueType” del segon mapeig correspondrà a una variable tipus DadesHI_, formada mitjançant l’element “struct”, i composta per totes les variables individualitzades de l’usuari en aquell període horari. Entre elles s’hi troba l’energia consumida restant horària individualitzada. Aquesta energia correspon a la diferència entre l’energia produïda horària individualitzada i l’energia consumida horària individualitzada pels usuaris autoconsumidors sense excedents la. També destaca la variable d’energia autoconsumida horària compartida, corresponent a una part d’energia repartida segons els consums dels usuaris que no supleixin la totalitat d’aquests amb autoconsum propi. Finalment s’hi inclouen el coeficient dinàmic horari individualitzat *ex post* beta de l’usuari en qüestió per aquell període temporal i el preu de l’electricitat en el mercat majorista aquella hora.


```

mapping( uint256 => mapping( uint256 => DadesHI_ ) ) public DadesHI; // primer uint256 es l'ID_user
//segon uint256 correspon a l'hora per al conjunt de dades horaries individualitzades
struct DadesHI_{
    uint256 ENPhi; //Energia produïda horaria individualitzada
    uint256 ENGhi; //Energia generada horaria individualitzada
    uint256 ENChi; //Energia consumida horaria individualitzada
    uint256 ENCrhi; //Energia consumida restant horaria individualitzada (només UA)
    uint256 ENCRhi; //Energia consumida de xarxa horaria individualitzada
    uint256 Eautci; //Energia autoconsumida horaria compartida
    uint256 Eauti; //Energia autoconsumida horaria individualitzada
    uint256 Betahi; //Coeficient de repartiment horari individualitzat
    uint256 PreuMMh; //Preu mercat majorista en la corresponent hora
}

```

Il·lustració 19: Declaració mapping DadesHI i struct DadesHI_ del contracte alfaList (Font: propia)

Posteriorment a la definició del mapeig DadesHI, es realitza una funció que calculi la Beta i els diferents valors per a cada usuari i període temporal concret per introduir-los en aquest. Per això es crea un “for” on una variable i parteixi de 0 fins a l’enter més proper sense incloure l’enter exacte de la variable “addressRegistryCount” i cada vegada que es realitzi el codi intern del “for” augmentarà la variable i en 1 unitat. Dins d’aquest es defineixen els tipus de les variables internes amb les quals es realitzaran els càlculs matemàtics i que posteriorment serviran per al finalitzar la funció introduir el conjunt de dades al “mapping” DadesHI. En alguns casos també se’ls hi dona un valor directament.

```

function setBeta() public {
    for(uint256 i = 0; i < addressRegistryCount; i++){

        // Declaració de variables internes per a la posterior introducció a DadesHI

        uint256 _ENPhi = Registre[i].EpWh;
        uint256 _ENGhi;
        uint256 _ENChi = Registre[i].EcWh;
        uint256 _ENCrhi;
        uint256 _ENCRhi;
        uint256 _Eautci;
        uint256 _Eauti;
        uint256 _Betahi;
        uint256 _PreuMMh = PreuMM[i];
    }
}

```

Il·lustració 20: Inici funció setBeta del contracte alfaList (Font: propia)

Un cop definides les variables, es despleguen 3 “if” diferents segons el cas en que ens trobem, un per a $\text{Alfa} \geq 100$, $100 > \text{Alfa} > 0$, o $\text{Alfa} = 0$. Es realitza d’aquesta forma ja que el llenguatge Solidity no permet decimals. Apart de Beta, també s’assignen valor a la resta de variables a introduir a DadesHI. El càlcul de Beta es lleugerament similar en els 3 casos però difereix en detalls.

En el primer “if”, quan $\text{Alfa} \geq 100$, s’assigna valor 0 a les variables d’energia restant, energia autoconsumida horària compartida, i energia utilitzada de la xarxa. Això es deu que aquests usuaris, ja supleixen la totalitat dels seus consums i no han de fer ús de cap d’aquests variables. Assumint que la seva producció és major als seus consums horaris, s’imposa que l’energia autoconsumida individualitzada sigui igual a l’energia consumida individualitzada. Per tan, es fa coincidir el valor de l’energia generada individualitzada amb el valor de l’energia consumida, calculant el coeficient Beta

mitjançant la divisió entre els consums horaris de l'usuari entre l'energia total generada en aquell període horari, tal i com s'exposa en l'Eq. 1.

$$\text{Alfa} \geq 100 \rightarrow \beta = 10^6 \cdot \frac{\text{ENChi}}{\text{ENGh}} \quad \text{Eq. 1}$$

```

if (Registre[i].Alfa >= 100 ){
    _ENCrhi = 0;
    _Betahi = (_ENChi * 1000000) / DadesH[Registre[i].Time].ENGh ;
    _ENGhi = (_Betahi * DadesH[Registre[i].Time].ENGh )/ 1000000 ;
    _Eautci = 0;
    _Eauti = _ENGhi;
    _ENCRhi = 0;
}

```

Il·lustració 21: Funció settBeta if per Alfa >= 100 del contracte alfaList (Font: propia)

En el segon "if", quan $0 < \text{Alfa} < 100$ es calcula l'energia consumida restant com la diferència entre l'energia consumida horària i l'energia produïda. La Beta horària es calcula de forma que l'usuari s'asseguri rebre la seva producció encara que no hagi generat excedents. Seguint aquesta premissa, es calcula Beta com la suma de l'energia produïda horària entre l'energia horària generada, i l'energia consumida restant per la quantitat total d'excedents entre el producte de l'energia consumida parcialment en aquell període(energia no autoconsumida directament) i l'energia generada horària. Seguint el dictat es dedueix el valor del coeficient dinàmic de repartiment a partir de l'Eq. 2.

$$0 < \text{Alfa} < 100 \rightarrow \beta = 10^6 \cdot \frac{\text{ENPhi}}{\text{ENGh}} + 10^6 \cdot \frac{\text{ENCrhi} \cdot \text{ET}}{\text{ENCPH} \cdot \text{ENGh}} \quad \text{Eq. 2}$$

Un cop obtinguda la Beta i conseqüentment l'energia generada horària, s'obre un "if" per determinar si l'energia generada individualitzada és major als consums horaris de l'usuari. Si es així, l'energia autoconsumida correspondrà a l'energia consumida horària per l'usuari. Si no, correspondrà a l'energia generada horària. Finalment es determina l'energia autoconsumida compartida com la diferència entre l'energia autoconsumida i l'energia produïda, i l'energia utilitzada de xarxa com la diferència entre l'energia consumida horària i l'energia autoconsumida.

```

if (0 <= Registre[i].Alfa && Registre[i].Alfa < 100 ){
    _ENCrhi = ENChi - _ENPhi;
    _Betahi = (_ENPhi * 1000000) / DadesH[Registre[i].Time].ENGh + (_ENCrhi * DadesH[Registre[i].Time].amountET * 1000000) / (DadesH[Registre[i].Time].ENCPH * DadesH[Registre[i].Time].ENGh) ;
    _ENGhi = (DadesH[Registre[i].Time].ENGh * _Betahi) / 1000000 ;
    if (_ENGhi >= _ENChi){
        _Eauti = _ENChi;
    } else {
        _Eauti = _ENGhi;
    }
    _Eautci = _Eauti - _ENPhi;
    _ENCRhi = _ENChi - _Eauti;
}

```

Il·lustració 22: Funció settBeta if per alfa entre 0 i 100 del contracte alfaList (Font: propia)

Finalment, en el tercer "if", quan Alfa = 0, s'implica que l'usuari al no tenir producció pròpia, l'energia consumida restant serà 0. Beta es calcula com la divisió entre l'energia consumida horària per la

quantitat total d'excedents entre el producte de l'energia consumida parcialment en aquell període(energia no autoconsumida directament) i l'energia generada horària. En aquesta situació, s'obté el valor de β deduït a partir de l'Eq. 3.

$$\text{Alfa} = 0 \rightarrow \beta = 10^6 \cdot \frac{\text{ENCrhi} \cdot \text{ET}}{\text{ENCPh} \cdot \text{ENGh}} \quad \text{Eq. 3}$$

Un cop obtinguda la Beta i consegüentment l'energia generada horària, s'obre un "if" per determinar si l'energia generada individualitzada és major als consums horaris de l'usuari. Si es així, l'energia autoconsumida correspondrà a l'energia consumida horària per l'usuari. Si no, correspondrà a l'energia generada horària. Finalment es determina l'energia autoconsumida compartida és igual a l'energia autoconsumida, ja que no t'he producció pròpia aquest tipus d'usuari, i l'energia utilitzada de xarxa com la diferència entre l'energia consumida horària i l'energia autoconsumida.

```

if (Registre[i].Alfa == 0 ){
    _ENCrhi = 0;
    _Betahi = (_ENChi * DadesH[Registre[i].Time].amountET * 1000000) / (DadesH[Registre[i].Time].ENCPh * DadesH[Registre[i].Time].ENGh);
    _ENGhi = (_Betahi * DadesH[Registre[i].Time].ENGh) / 1000000 ;
    if(_ENGhi >= _ENChi){
        _Eauti = _ENChi;
    } else {
        _Eauti = _ENGhi;
    }
    _Eautci = _Eauti;
    _ENCrhi = _ENChi - _Eauti;
}

```

Il·lustració 23: Funció settBeta if Alfa = 0 del contracte alfaList (Font: propia)

Un cop realitzat el càlcul de Beta, s'enllaça en el mapeig DadesHI, utilitzant com a "_KeyType" el DNI de l'usuari i el codi de temps (1-8760) de l'hora corresponent.

```

DadesHI[Registre[i].ID_user][Registre[i].Time] = DadesHI( _ENPhi, _ENGhi, _ENChi, _ENCrhi, _ENCrhi, _Eautci, _Eauti, _Betahi, _PreuMWh);
}
}

```

Il·lustració 24: Funció settBeta introducció de dades al mapping DadesHI del contracte alfaList (Font: propia)

Finalment es creem una funció getDadeshi per tal de poder observar ordenat, segons el registres realitzats, el diferents valors introduïts al "mapping" DadesHI. Per això, després del nom de la funció i tancar els parèntesis d'aquesta sense cap element dins, ja que no s'han d'introduir dades externes per tal d'aconseguir-ho, posem que la funció sigui "public". En aquest cas, degut a que la quantitat de variables internes del contracte no permet la seva compilació, es soluciona posant aquest apartat en visibilitat "private". També s'escriu la paraula "view", ja que el que s'estarà fent serà llegir dades escrites prèviament a la blockchain, i també s'hi posarà un "returns()". A l'interior d'aquest hi aniran 9 array de "uint256", un conjunt d'enters numèrics, i es citarà la paraula "memory" per a cada un d'ells per tal de poder recuperar les dades introduïdes en la "blockchain".

A partir d'aquí s'obren les claus, necessàries en totes les funcions, i es defineixen els array de "uint256", mencionats en el paràgraf anterior, d'una llargada igual al valor de "addressRegistryCount" i amb el seu nom corresponent a cada variable de DadesHI. Un cop definit iniciem un "for" per a valors d'una variable interna i de 0 fins a al valor més gran però o no igual a "addressRegistryCount" i que augmenti en 1 unitat cada vegada que acaba un cicle. Dins d'aquest "for" es defineixen els diferents valors que tindran els array per a les diferents posicions i, on aquests correspondran al valor de del mapeig DadesHI amb els "_KeyType" ID_user i Time del mapeig Registre.

Finalment, fora del "for", l'element "return", retornarà tots els arrays definits i emplenats amb els valors de DadesHI.

```
function getDadeshi() private view returns (uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory){
    uint256[] memory enphi = new uint256[](addressRegistryCount);
    uint256[] memory enghi = new uint256[](addressRegistryCount);
    uint256[] memory enchi = new uint256[](addressRegistryCount);
    uint256[] memory encrhi = new uint256[](addressRegistryCount);
    uint256[] memory encrhi = new uint256[](addressRegistryCount);
    uint256[] memory eautci = new uint256[](addressRegistryCount);
    uint256[] memory eauti = new uint256[](addressRegistryCount);
    uint256[] memory betta = new uint256[](addressRegistryCount);
    uint256[] memory preuWh = new uint256[](addressRegistryCount);
    for(uint256 i = 0; i < addressRegistryCount; i++){
        enphi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENPHi;
        enghi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENGRi;
        enchi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENCHi;
        encrhi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENCRhi;
        encrhi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENCRhi;
        eautci[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Eautci;
        eauti[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Eauti;
        betta[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Betahi;
        preuWh[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].PreuWh;
    }
    return (enphi, enghi, enchi, encrhi, encrhi, eautci, eauti, betta, preuWh);
}
```

Il·lustració 25: Funció getDadeshi del contracte alfaList (Font: propia)

6.3. Facturació

Inicialment es defineix la llicència i la versió de Solidity en la primera línia. Seguidament es defineix el nom i el contingut del contracte el qual estarà limitat per la clau que hi haurà després del nom d'aquests i la clau de l'última línia del contracte.

A continuació declarem variables que serviran pel llarg del contracte, entre les quals hi haurà dos variables tipus "address", on al realitzar el deploy del contracte si introduiran les "address" dels dos contractes previs, anomenats alfa_List i Registre_usuaris. També s'hi ha posat una variable tipus "uint256", anomenada CostP amb la qual introduïrem el cost del terme de potència fixa al realitzar el "deploy" a la "blockchain", per tal de realitzar el càlcul posterior de forma més senzilla.

A continuació indiquem dins el constructor que se l'hi entraran aquestes 3 variables escrites de forma externa i el tipus de cada una, i que les assigni a les variables internes pròpies del contracte. D'aquesta manera assignem una variable local del "constructor" a una variable interna del contracte.

```

//SPDX-License-Identifier: MIT
pragma solidity >=0.5.0;

contract Facturacio{
    address public alfa_List;
    address public Registre_usuaris;
    uint256 public CostP;

    constructor(address _addressC, address _addressD, uint256 _CostP) {
        alfa_List = _addressC;
        Registre_usuaris = _addressD;
        CostP = _CostP;
    }
}

```

Il·lustració 26: Inici declaració de variables d'estat i funció inicialitzadora del contracte Facturacio (Font: propia)

Seguidament es realitza un “mapping” o llistat amb 2 claus, 2 dades o valors que ens permetran emmagatzemar i classificar les dades de la factura de cada usuari segons el DNI d'aquest, primera clau, i el número de la factura, segona clau.

Dins aquest nou tipus Factura creat a partir de l'element “struct”, que permet crear nous tipus de variables, s'han agrupat totes les dades que contindria una factura: des de l'identificador de l'usuari(Identificador), el seu DNI (DNI_usuari), la “address” de la “wallet” o moneder des don es realitzarà el pagament d'aquesta (U_address), la potència en kW contractada (kWc), el número de factura corresponent (numfact), la quantitat d'energia utilitzada provinent de la xarxa en kWh (kWhcr), la quantitat d'energia autoconsumida de forma compartida (kWhautc), la quantitat d'energia autoconsumida total (kWhautT), la quantitat total d'energia consumida (kWhCT), la quantitat d'energia excedentària si s'és productor (kWhe), el preu a pagar referent al terme de potència en €/kW-dia (PreukWc), el preu promig a pagar referent a l'energia autoconsumida de forma compartida (PreukWhautc), el preu promig a que es paga l'energia excedentària dels productors de la comunitat (Preukwhe), i el preu promig a pagar en concepte d'energia provinent de la xarxa de transport i distribució (PreukWhcr).

Finalment es defineixen les variables que contindran l'import a pagar corresponent als diferents termes de la factura: l'import referent al terme de potència contractada en cèntims d'euro (cEurPC), l'import referent al terme d'energia utilitzada per l'usuari en cèntims d'euro (cEurEU), l'import referent al terme de l'impost sobre l'energia elèctrica en cèntims d'euro (cEurIE), l'import referent a altres termes de la factura en cèntims d'euro (cEurOther), l'import referent a l'IVA de la factura en cèntims d'euro (cEurIVA), l'import total de la factura en cèntims d'euro (cEurAll).

```

mapping(uint256 => mapping (uint256 => Factura)) public Factures;//1ª Clau DNI_usuari, 2ª clau Identificador Factura
struct Factura { //Totes les unitats monetaries estaran en cèntims d'Euro.
    uint256 Identificador;
    uint256 DNI_usuari;
    address U_address;
    uint32 kWc;
    uint256 numfact;
    uint256 kWhcr;
    uint256 kWhautc;
    uint256 kWhautI;
    uint256 kWhCT;
    uint256 kWhc;
    uint256 PreukWc;//Cost potència contractada.
    uint256 PreukWhautc;//Preu promig de l'energia autoconsumida compartida.
    uint256 PreukWhe;//Preu promig de l'energia excedentaria.
    uint256 PreukWhcr;// Promig total energia consumida xarxa.
    uint256 cEurPC;
    uint256 cEurEU;
    uint256 cEurIE;//5.113% del terme de potència i Energia consumida total.
    uint256 cEurOther;//Terme per energia compartida a la comunitat.
    uint256 cEurIVA;//IVA 10%.
    uint256 cEurAll;
}

```

Il·lustració 27: Declaració mapping Factures i struct Factura del contracte Facturacio (Font: propia)

A continuació es realitza una funció que agafarà dades recollides en els altres 2 “*smart-contracts*”, mitjançant la seva “*address*” definida prèviament, les utilitzarà per realitzar els càlculs de les diferents variables necessàries per obtenir l’import total de la factura, i s’enregistraran aquests valors a la “*blockchain*” introduint-los al “*mapping*” Factures.

Inicialment es declara el nom de la funció, la visibilitat pública d’aquesta, els claudàtors inicial, situat just després de la visibilitat, i final, situat a la segona columna després de tot el codi corresponent a la funció.

Seguidament es recupera el valor identificadors_contadors, mitjançant l’ús del mètode intern de Solidity `.call` i la “*address*” del contracte `Registre_usuaris` definida prèviament. Aquest mètode permet utilitzar dades, funcions, elements, etc, d’altres “*smart-contracts*” utilitzant la seva “*address*”, i al seu interior s’utilitza “*abi.encodeWithSignature()*” el qual es una drecera ofertada per Solidity que transforma en “*bytes*” el “*Hash*” de la funció, element o variable que li introduïm del corresponent contracte. En aquest cas, crida a la variable `identificadors_contadors`, la qual no necessita que se li introdueixi cap variable perquè funcioni a diferència d’altres funcions o elements. Un cop obtingut la variable tipus “*bytes*” `Rcount` es descodifica a una variable tipus “*uint256*”, anomenada `_ldcount`, per poder operar amb ella.

A continuació dins la mateixa funció, es realitza un “*for*” per analitzar tots els valors de una variable tipus “*uint256*” des de 0 fins el valor de la variable `_ldcount`, augmentat de forma progressiva en 1 unitat el seu valor al completar el cicle. Per a cada valor de “*i*” s’extreuen dades del llistat realitzat en l’*smart-contract* `Registre_usuaris` dins el mapeig anomenat `Contadors`, i amb la clau corresponent a la variable “*i*”. Es realitzarà aquest procés mitjançant el mètode utilitzat anteriorment per agafar el valor de la variable `identificadors_contadors` de l’*smart-contract* `Registre_usuaris`. Seguint amb aquest procés, es descodifiquen en els tipus de variables corresponents tots els elements que conté la variable creada mitjançant un “*struct*” anomenat `SmartM` que és la variable que ens retorna el “*mapping*”

Contadors. D'aquesta forma s'analitzaran i utilitzaran les dades de tots els usuaris registrats en el primer smart-contract.

```
function setfactures() public{
    (bool success, bytes memory Rcount) = Registre_usuaris.call(abi.encodeWithSignature("identificadors_contadors"));
    require(success, "Error");
    (uint256 _idcount) = abi.decode(Rcount, (uint256));

    for(uint256 i = 0; i < _idcount; i++){
        (bool success1, bytes memory data) = Registre_usuaris.call(abi.encodeWithSignature("contadors[uint256]",i));
        require(success1, "Error");
        (uint256 _id, string memory _tipus1, uint256 _DNI_usuari, address _U_Address, uint32 _kWhp1, uint32 _kWhc) = abi.decode(data, (uint256, string, uint256, address, uint32, uint32));

        uint256 _numfact;
        uint256 _kWhc;
        uint256 _kWhautc;
        uint256 _kWhauti;
        uint256 _kWhCF;
        uint256 _kWhc;
        uint256 _PreukWhautc;
        uint256 _PreukWhc;
        uint256 _PreukWhc;
        uint256 _cEurPC;
        uint256 _cEurE;
        uint256 _cEurIE;
        uint256 _cEurOther;
        uint256 _cEurIVA;
        uint256 _cEurAll;
        uint256 _cEurautci;
        uint256 _cEurR;
    }
}
```

Il·lustració 28: Recollida de dades del contracte Registreusuaris dins la funció setfactures del contracte Facturacio (Font: propia)

Seguidament es declaren les variables internes necessàries per introduir els seus valors a les variables contingudes dins la variable creada prèviament anomenada Factura.

Un cop declarades, s'obre un nou "for", el qual analitzarà totes les hores del període de facturació, any, etc, essent el valor 24 modificable. Conjuntament amb aquest valor i el DNI de l'usuari importat anteriorment, s'importaran les dades registrades en el "mapping" DadesHI segons el procés utilitzat prèviament. Un cop descodificats tots els valors importats als seus corresponents tipus de variables s'inicien els càlculs per determinar la quantia total en el període de facturació de les diferents variables. Dins el mateix "for", es defineix una variable "uint256" anomenada CostkWceh, corresponent al preu de compravenda en aquell període horari per a l'energia excedentària dels productors i a l'energia d'autoconsum compartit de la resta d'usuaris sense excedents.

A continuació es realitza un "if" per determinar els períodes horaris en que es produeixin excedents i tenir present l'import de la venda d'aquest segons el respectiu preu calculat.

Finalment, es calcula l'import de l'energia consumida per l'usuari, realitzant la suma dels producte de l'energia autoconsumida compartida individualitzada i l'energia utilitzada de xarxa horària, pels seus corresponents preus establerts prèviament. Aquest valor es va acumulant per un mateix usuari durant els diferents períodes de temps. També, a mètode informatiu, es realitza amb el mateix procés que l'anterior únicament la suma dels productes entre el cost de l'energia autoconsumida compartida i la quantia d'aquesta. Finalment es calcula d'igual forma als anteriors el cost de l'energia extreta de la xarxa i es va sumant als anteriors pera cada usuari.

```

for(uint256 t = 0; t < 24; t++){
    (bool success2, bytes memory data1) = alfa_List.call(abi.encodeWithSignature("DadesH[uint256][uint256]", _DNI_usuari, t));
    require(success2, "Error");
    (uint256_ENPhi, uint256_ENGhi, uint256_ENChI, uint256_ENCRhi, uint256_ENCRhi, uint256_Eautci, uint256_Eauti, uint256_Betahi, uint256_PreuMMh) =
    abi.decode(data1, (uint256, uint256, uint256, uint256, uint256, uint256, uint256, uint256));
    {
        _kWhcr += _ENCRhi;
        _kWhautc += _Eautci;
        _kWhautT += _Eauti;
        _kWhCT += _ENChI;
    }
    uint256 Costkwech = (7 + _PreuMMh)/2;
    {
        if(_ENPhi > _ENChI){
            _kWe += _ENPhi - _ENChI;
            _cEurOther -= Costkwech*( _ENPhi - _ENChI);
        }
        else{
            _kWe += 0;
            _cEurOther += 0;
        }
        _cEurEU += Costkwech* _Eautci + _PreuMMh*_ENCRhi;
        _cEurautci += Costkwech* _Eautci;
        _cEurR += _PreuMMh*_ENCRhi;
    }
}

```

Il·lustració 29: recollida de dades de alfaList dins funció setBeta i càlcul dels termes de la factura del contracte Facturacio (Font: propia)

Un cop finalitzat el cicle, es determinen els preus promig dels diferents imports del terme d'energia. També es calcula el total dels diferents imports, realitzant la suma d'aquests per obtenir l'import total de la factura.

Finalment s'introdueixen tots els valors de la variable Factura en el "mapping" amb doble clau Factures, utilitzant el DNI i el número de factura com a claus.

```

{
    _PreukWhautc = _cEurautci/_kWhautc;
    _PreukWhcr= _cEurR/_kWhcr;
    _PreukWe = _cEurOther/_kWe;
    _cEurPC = CostP*_kWhpc;
    _cEurIE = ((_cEurEU + _cEurPC) * 5113)/100000;
    _cEurIVA = ((_cEurEU + _cEurPC + _cEurIE + _cEurOther)*10)/100;
    _cEurAll = _cEurEU + _cEurPC + _cEurIE + _cEurOther + _cEurIVA;
}

Factures[_DNI_usuari][_numfact] =Factura(_id, _DNI_usuari, _U_Address, _kWhpc, _numfact, _kWhcr, _kWhautc, _kWhautT, _kWhCT, _kWe, CostP,
_PreukWhautc, _PreukWe, _PreukWhcr, _cEurPC, _cEurEU, _cEurIE, _cEurOther, _cEurIVA, _cEurAll);
}

```

Il·lustració 30: Càlcul de termes de la factura i introducció d'aquests al mapping Factures del contracte Facturacio (Font: propia)

Finalment es realitza una funció mitjançant la qual els usuaris poden pagar l'import de la factura, introduint el DNI de l'usuari registrat i el número de factura corresponent. També hauran de verificar el valor de la transacció prèviament perquè si no es igual al valor calculat no es podrà realitzar.

```

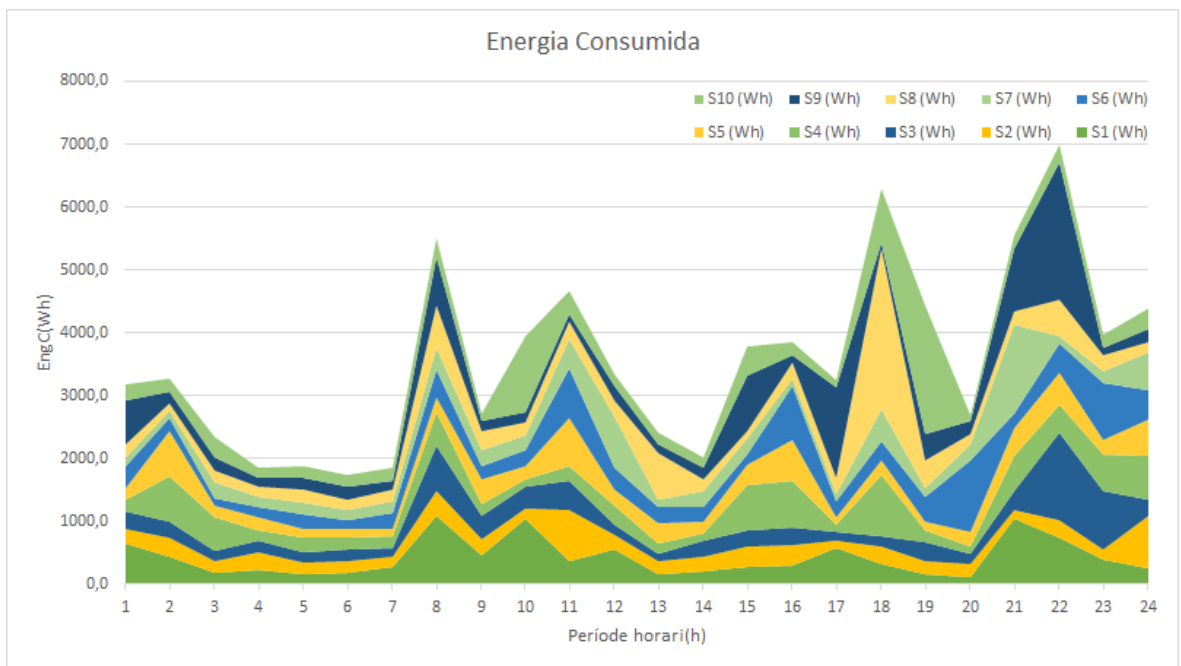
function PayFact(uint256 _DNI_usuari, uint256 _numfact) public payable{
    require(msg.value == Factures[_DNI_usuari][_numfact].cEurAll*1000000);
}

```

Il·lustració 31: Funció PayFact del contracte Facturacio (Font: propia)

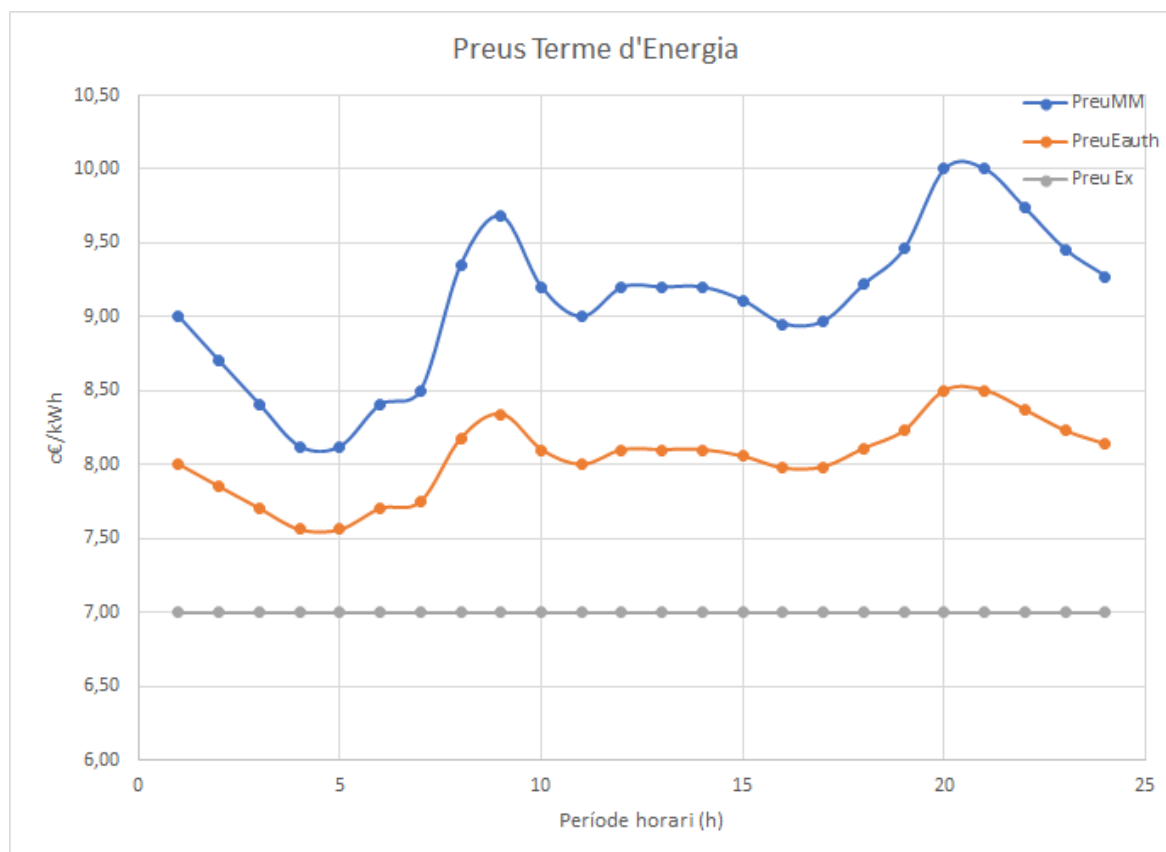
7. Simulació de la Dapp Dissenyada

A continuació, per tal de poder realitzar una petita comparació respecte altres escenaris, es presenta una simulació realitzada en 3 escenaris diferents corresponents al càlcul d'una factura de 24h amb diferents premisses, però els quals parteixen de les mateixes dades base. Aquestes consisteixen de les corbes de consum i producció, el preu del mercat majorista, el preu dels excedents, la potència contractada i el cost associat a aquesta. Conjuntament també són homogenis l'impost a l'electricitat i l'IVA.



Il·lustració 32. Consums superposats dels usuaris d'estudi (Font: pròpia)

En la Il·lustració 32, s'observen els consums horaris utilitzats en l'estudi, i les seves variacions al llarg del temps d'anàlisi. Al estar superposades, també s'aprecia els consums del conjunt d'aquests com a comunitat.



Il·lustració 33. Preus en c€/kWh referents als diferents conceptes dins el terme d'energia (Font: propia)

En la Il·lustració 33, s'exposen de forma visual els preus utilitzats per a la simulació dels 3 escenaris d'estudi, considerant els mateixos per a tots ells. PreuMM, PreuEauth i PreuEX estan en cèntims d'euro per quilowatt hora per tal d'una millor percepció dels resultats obtinguts degut el curt període d'anàlisi. Els valors del PreuMM han set extrets de l'operador del mercat ibèric d'electricitat (61).

7.1. Escenari 1

En aquest escenari, es presenta una situació bàsica on tots els consumidors són únicament consumidors i no participen d'una comunitat energètica local. D'aquesta forma únicament es calcula la factura en funció de l'energia consumida (Eng_{Ch}), el preu associat a aquesta segons el mercat majorista ($PreuMM_h$), i la potència contractada i el cost que li correspon. El $PreuMM_h$ és extret de la base de dades de l'operador del mercat ibèric d'energia (OMIE) de forma que les dates dels consums i producció es corresponguin amb les dels preus utilitzats. Un cop realitzat aquest càlculs es suma al terme de potència ($CostTP$) i al terme d'energia ($CostTE$) del període d'estudi (24 h), i se'ls hi aplica l'impost sobre l'electricitat (10%) i l'IVA (21%).

$$CostTE_i = \sum_{h=1}^{24} (EngC_{hi} \cdot PreuMM_h) \quad \text{Eq. 4}$$

$$CostTP_i = PotC_i \cdot PreuPC_d \cdot Dies \quad \text{Eq. 5}$$

$$Total_i = (CostTE_i + CostTP_i) \cdot 1,1 \cdot 1,21 \quad \text{Eq. 6}$$

	U1	U2	U3	U4	U5
<i>CostTE_i</i>	94,48 c€	60,69 c€	74,52 c€	81,98 c€	68,92 c€
Total	192,30 c€	147,33 c€	165,74 c€	175,67 c€	158,28 c€
	U6	U7	U8	U9	U10
<i>CostTE_i</i>	84,90 c€	68,38 c€	82,16 c€	94,70 c€	81,27 c€
Total	179,55 c€	157,56 c€	175,90 c€	192,60 c€	174,72 c€

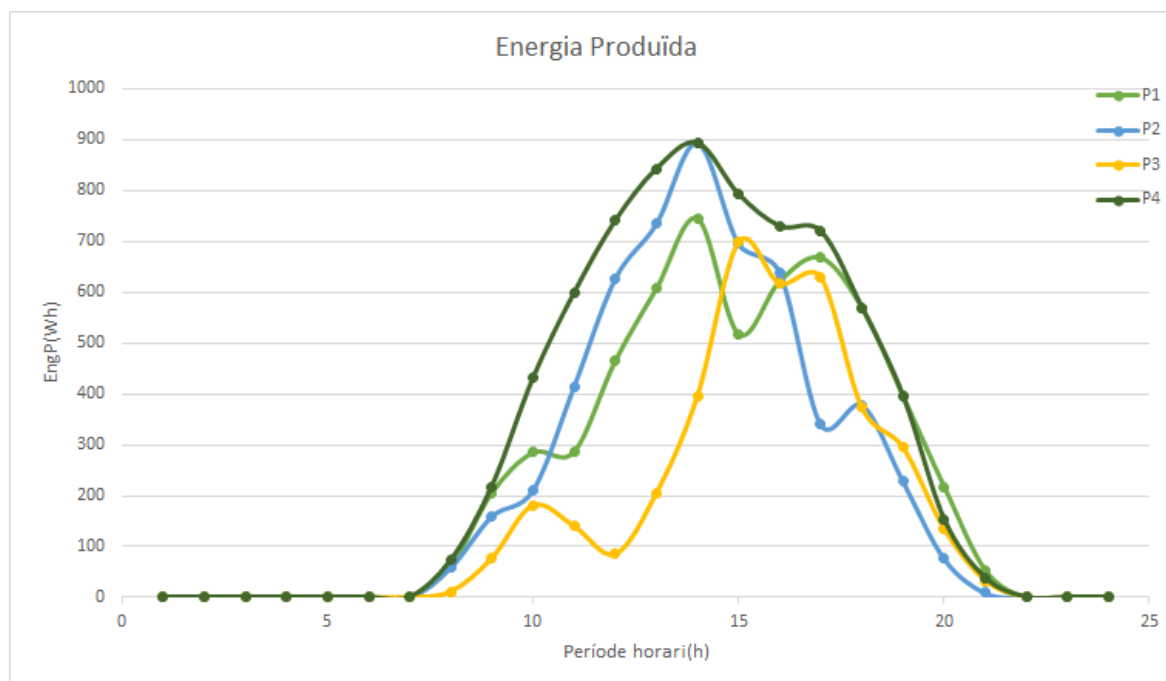
Taula 3. Resultats econòmics de la simulació de l'escenari 1 (Font: propia)

Seguin la premissa anterior, i un cop realitzats els càlculs, partint de les dades comentades anteriorment, s'obtenen els resultats de la Taula 3. S'observa el cost corresponent al terme d'energia, obtingut a partir de l'Eq. 4, de cada usuari d'estudi durant les 24 h d'aquest període. També s'hi contempla el cost total corresponent a la base imposable, suma dels termes d'energia (Eq. 4) i potència (Eq. 5), l'aplicació del 10% de l'impost elèctric i el 21 % respecte la base imposable referent a l'IVA. Així, en aquest escenari base, es contempla el càrrec elèctric d'un dia per tal de tenir un punt de comparativa respecte el següents escenaris plantejats.

7.2. Escenari 2

En el segon escenari, es planteja de manera que 4 dels consumidors a estudiar tinguin instal·lacions solars a les seves cobertes produint energia per a si mateixos però sense esta associats a cap

comunitat, ni entre els productors, ni amb els consumidors.



Il·lustració 34. Energia produïda pels usuaris amb instal·lacions renovables (Font: Propia)

En la Il·lustració 34, s'aprecia la producció d'energia solar utilitzada per a les simulacions de l'escenari 2 i 3 dels 4 usuaris productors.

Seguint els fets indicats, es descompta l'energia produïda dels consums dels productors, i en els casos on es produeixen excedents es venen a la xarxa a un preu d'excedents (PreuEX) igual a 7 c€/kWh. Aquest és el preu de referència segons les comercialitzadores per a les instal·lacions acollides a la modelitat d'autoconsum amb excedents i compensació. Si amb l'energia produïda horària, no es supeix el consum horari de l'usuari, es compra l'energia a la xarxa a preu de mercat. Finalment el cost total individualitzat dels usuaris surt de l'Eq. 9, utilitzant el nou import del terme d'energia calculat mitjançant l'Eq. 7 i l'Eq. 8.

$$\text{Si } EngC_{hi} - EngP_{hi} > 0 \rightarrow CostTE_{hi} = (EngC_{hi} - EngP_{hi}) \cdot PreuMM_h \quad \text{Eq. 7}$$

$$\text{Si } EngC_{hi} - EngP_{hi} \leq 0 \rightarrow CostTE_{hi} = (EngC_{hi} - EngP_{hi}) \cdot PreuEX \quad \text{Eq. 8}$$

$$CostTE_i = \sum_{h=1}^{24} CostTE_{hi} \quad \text{Eq. 9}$$

	U1	U2	U3	U4	U5
$CostTE_i$	46,83 c€	16,31 c€	42,47 c€	23,49 c€	68,92 c€
Total	128,88 c€	88,25 c€	123,08 c€	97,82 c€	158,28 c€
	U6	U7	U8	U9	U10
$CostTE_i$	84,90 c€	68,38 c€	82,16 c€	94,70 c€	81,27 c€
Total	179,55 c€	157,56 c€	175,90 c€	192,60 c€	174,72 c€

Taula 4. Resultats econòmics de la simulació de l'escenari 2 (Font: propia)

Seguint la premissa indicada, i partint de les dades inicials declarades, es pot observar en la Taula 4 clarament com els usuaris productors veuen reduït dràsticament el seu cost en el terme d'energia consumida, degut a la instal·lació de panells fotovoltaics per a la producció d'energia elèctrica mitjançant energies renovables. Aquest fet es veu reflectit en l'import total que se li correspon a cada usuari.

7.3. Escenari 3

Finalment en l'últim escenari, es planteja que el conjunt dels usuaris ha format una comunitat energètica, amb l'objectiu de compartir els excedents d'aquells usuaris que compten amb una instal·lació de producció d'energia renovable solar. Per dur a terme aquest objectiu, s'implementa una aplicació descentralitzada de disseny propi desenvolupada sobre tecnologia "blockchain", mitjançant la qual es calculen coeficients de repartiment dinàmic *ex post*, aplicats posteriorment a l'energia generada pel conjunt de la comunitat. Posteriorment quan l'usuari vulgui realitzar el pagament de la factura corresponent, es calcula l'import respectiu a través de les dades registrades i els coeficients calculats. Finalment, l'usuari ha de firmar la transacció de l'import calculat pel programa, verificant que l'import designat correspongui al seu usuari, i al firmar la transacció, es realitzarà el pagament en criptomoneda.

Seguint així amb les premisses realitzades, es calcula el cost associat al terme d'energia segons diferents factors. Com a prioritat, per tal de potenciar l'entrada d'usuaris productors en la comunitat, únicament es comparteix amb la resta d'usuaris aquella energia que no consumeixi el propi productor, o sigui l'excedentària (Ex_{hi}), com es dedueix de l'Eq. 10. Aquesta energia, serà producte d'una compravenda, entre el productor i la resta d'usuaris que tinguin uns consums majors a la seva producció d'energia, a un preu promig entre el $PreuMM_h$, i el $PreuEX$. Un cop subministrada la totalitat dels consums, la resta d'energia serà abocada a la xarxa a un preu de venda $PreuEX$, el qual es dividirà el benefici obtingut per aquesta energia en funció de l'energia produïda per l'usuari entre l'energia produïda total, de forma que es reparteixi equitativament entre ells.

$$\mathbf{Si} \ EngC_{hi} - EngP_{hi} < 0 \rightarrow Ex_{hi} = EngC_{hi} - EngP_{hi} \quad \mathbf{Eq. 10}$$

En el moment en que l'usuari productor no tingui excedents, i per contra no sigui possible cobrir els seus consums en aquell període horari amb producció pròpia, utilitzaran l'energia excedentària dels productors de forma equitativa amb la resta d'usuaris consumidors o productors en la mateixa situació. En utilitzar aquesta energia, es realitzarà la compravenda d'aquesta amb els usuaris productors d'aquests excedents.

Seguint el descrit, es dedueix en les Eq. 11Eq. 12Eq. 13Eq. 14, el cost referent al terme d'energia dels usuaris que en aquell moment estiguin produint energia segons la seva situació, i la situació del conjunt de la comunitat.

$$\mathbf{Si} \ EngC_{hi} - EngP_{hi} - EngC_{aut} > 0 \rightarrow$$

$$CostTE_{hi} = (EngC_{hi} - EngP_{hi} - EngC_{aut}) \cdot PreuMM_h + \quad \mathbf{Eq. 11}$$

$$EngC_{aut} \cdot \frac{PreuMM_h - PreuEX}{2}$$

$$\mathbf{Si} \ EngC_{hi} - EngP_{hi} - EngC_{aut} \leq 0 \ \& \ EngC_{hi} - EngP_{hi} > 0 \rightarrow$$

$$CostTE_{hi} = EngC_{aut} \cdot \frac{PreuMM_h - PreuEX}{2} \quad \mathbf{Eq. 12}$$

$$\mathbf{Si} \ EngC_{hi} - EngP_{hi} < 0 \ \& \ ConsumT_h - ProdT_h > 0 \rightarrow$$

$$CostTE_{hi} = (EngC_{hi} - EngP_{hi}) \cdot \frac{PreuMM_h - PreuEX}{2} \quad \mathbf{Eq. 13}$$

$$\text{Si } EngC_{hi} - EngP_{hi} < 0 \ \& \ ConsumT_h - ProdT_h < 0 \rightarrow$$

$$CostTE_{hi} = (EngC_{hi} - EngP_{hi}) \cdot \left(1 - \frac{\sum_{i=1}^{10} Ex_{hi}}{ConsumT_h - ProdT_h}\right) \cdot \frac{PreuMM_h - PreuEX}{2} +$$

Eq. 14

$$(EngC_{hi} - EngP_{hi}) \cdot \left(\frac{\sum_{i=1}^{10} Ex_{hi}}{ConsumT_h - ProdT_h}\right) \cdot PreuEX$$

Finalment per als usuaris que no tinguin una producció pròpia d'energia s'utilitzarà l'energia excedentària dels productors de forma equitativa amb la resta d'usuaris en la mateixa situació i amb els usuaris productors que no hagin pogut cobrir la totalitat de la seva demanda amb la producció pròpia. D'aquesta forma, i seguint les Eq. 15Eq. 16Eq. 17, es dividirà l'energia excedentària, segons el consum no suplert directament de l'usuari, mitjançant producció pròpia, respecte la totalitat dels consums restants dels productors sense excedents i dels consums totals dels usuaris sense producció. Aquesta energia excedentària sotmesa a compravenda, tindrà un preu intermedi entre el PreuMM_h, i el PreuEX. Si en aquests casos l'energia autoconsumida de forma compartida no és suficient per suplir la totalitat dels consums de l'usuari, s'utilitzarà energia de la xarxa a preu PreuMM_h. Això voldrà dir que en el conjunt de la comunitat no s'haurà produït suficient energia com per assumir la totalitat dels consums del usuaris associats.

$$\text{Si } \sum_{i=1}^{10} Ex_{hi} = 0 \rightarrow CostTE_{hi} = EngC_{hi} \cdot PreuMM_h$$

Eq. 15

$$\text{Si } \sum_{i=1}^{10} Ex_{hi} < 0 \ \& \ ConsumT_h - ProdT_h > 0 \rightarrow$$

$$CostTE_{hi} = (EngC_{hi} + \frac{(\sum_{i=1}^{10} Ex_{hi}) \cdot EngC_{hi}}{EngP_h}) \cdot PreuMM_h -$$

Eq. 16

$$\frac{(\sum_{i=1}^{10} Ex_{hi}) \cdot EngC_{hi}}{EngP_h} \cdot \frac{PreuMM_h - PreuEX}{2}$$

$$Si \sum_{i=1}^{10} Ex_{hi} < 0 \ \& \ ConsumT_h - ProdT_h < 0 \rightarrow$$

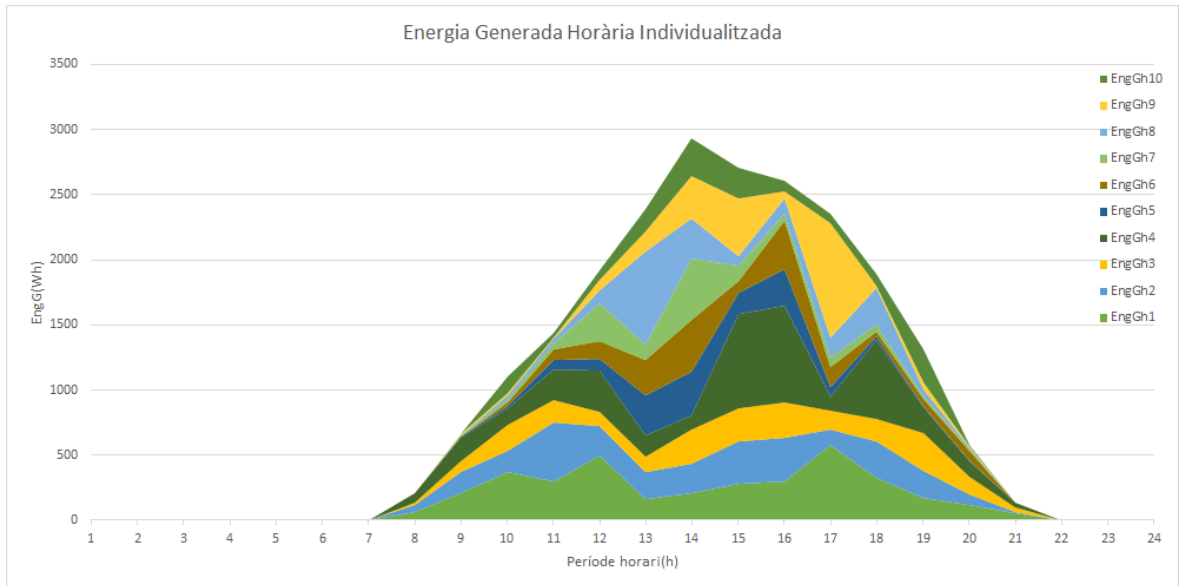
Eq. 17

$$CostTE_{hi} = EngC_{hi} \cdot \frac{PreuMM_h - PreuEX}{2}$$

El repartiment d'aquesta energia excedentària està vinculat directament a l'energia generada horària individualitzada. Aquesta és el producte del coeficient dinàmic *ex post* per l'energia generada total. Segons la tipologia de l'usuari, l'energia generada horària individualitzada tindrà un valor diferent, en funció de l'energia autoconsumida pròpia, l'energia autoconsumida compartida, o els excedents finals que hagin pogut sorgir. D'aquesta forma es genera un coeficient per cada hora i usuari que la suma de la seva totalitat en el mateix període horari ha de ser 1 o 100%, assegurant així que en cap moment s'està comerciant més energia de la que genera la comunitat. En la Taula 5, s'exposen els diferents valors calculats dels coeficients dinàmics *ex post* i la suma total d'aquests.

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}	$\Sigma\beta$
1	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
2	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
3	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
4	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
5	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
6	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
7	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
8	30,24%	28,78%	5,37%	35,61%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%
9	31,83%	24,47%	12,43%	27,53%	1,00%	0,54%	0,67%	0,78%	0,40%	0,36%	100,00%
10	33,33%	14,72%	17,99%	11,33%	1,91%	2,73%	2,17%	2,21%	1,55%	12,07%	100,00%
11	20,54%	31,67%	11,91%	16,02%	5,50%	5,60%	3,24%	2,11%	0,82%	2,59%	100,00%
12	26,07%	11,71%	5,78%	16,28%	5,03%	6,66%	15,40%	5,19%	4,19%	3,70%	100,00%
13	6,96%	8,49%	4,82%	7,07%	12,87%	11,28%	4,77%	30,25%	6,39%	7,10%	100,00%
14	7,24%	7,44%	8,97%	3,93%	11,28%	13,65%	15,93%	10,60%	11,03%	9,92%	100,00%
15	10,27%	12,03%	9,42%	26,95%	5,81%	3,39%	4,46%	2,49%	16,62%	8,55%	100,00%
16	11,47%	12,64%	10,67%	28,31%	10,91%	14,26%	1,94%	4,61%	1,94%	3,24%	100,00%
17	24,63%	4,89%	6,00%	4,89%	2,98%	6,63%	2,98%	6,63%	37,29%	3,10%	100,00%
18	17,40%	14,45%	9,05%	32,47%	1,33%	1,73%	3,06%	14,90%	0,68%	4,92%	100,00%
19	12,91%	15,83%	22,46%	14,65%	1,24%	3,78%	1,37%	4,24%	3,95%	19,57%	100,00%
20	19,83%	14,45%	23,43%	19,83%	2,53%	12,13%	2,74%	1,80%	2,03%	1,23%	100,00%
21	40,30%	6,72%	24,63%	28,36%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%
22	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
23	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
24	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%

Taula 5. Coeficients dinàmics *ex post* de l'escenari 3 (Font: Propia)



Il·lustració 35. Energia Generada Horària Individualitzada de l'escenari 3 (Font: Propia)

En la Il·lustració 35, s'exposa visualment l'energia generada horària individualitzada atribuïda a cada usuari, on el conjunt total d'aquestes en cada període, es l'energia horària produïda per la comunitat.

Finalment, es suma la totalitat dels costos del terme d'energia per a cada hora dels diferent usuaris, segons l'Eq. 18, i conjuntament amb l'import corresponent al terme de potència, s'apliquen els impostos corresponents, obtenint els resultats mostrats en la Taula 6.

$$CostTE_i = \sum_{h=1}^{24} CostTE_{hi} \quad \text{Eq. 18}$$

	U1	U2	U3	U4	U5
$CostTE_i$	44,51 c€	13,69 c€	40,69 c€	20,01 c€	67,57 c€
Total	125,80 c€	84,77 c€	120,71 c€	93,18 c€	156,49 c€
	U6	U7	U8	U9	U10
$CostTE_i$	83,28 c€	67,20 c€	80,25 c€	92,72 c€	79,80 c€
Total	177,39 c€	155,99 c€	173,37 c€	189,97 c€	172,77 c€

Taula 6. Resultats econòmics de la simulació de l'escenari 3 (Font: propia)

7.4. Comparativa Resultats

S'han ajuntat els resultats econòmics obtinguts dels diferents escenaris per fer més senzilla la seva comparació en la Taula 7.

	Escenari	$CostTE_i$	Total		Escenari	$CostTE_i$	Total
U1	E1	94,48 c€	192,30 c€	U6	E1	84,90 c€	179,55 c€
	E2	46,83 c€	128,88 c€		E2	84,90 c€	179,55 c€
	E3	44,51 c€	125,80 c€		E3	83,28 c€	177,39 c€
U2	E1	60,69 c€	147,33 c€	U7	E1	68,38 c€	157,56 c€
	E2	16,31 c€	88,25 c€		E2	68,38 c€	157,56 c€
	E3	13,69 c€	84,77 c€		E3	67,20 c€	155,99 c€
U3	E1	74,52 c€	165,74 c€	U8	E1	82,16 c€	175,90 c€
	E2	42,47 c€	123,08 c€		E2	82,16 c€	175,90 c€
	E3	40,69 c€	120,71 c€		E3	80,25 c€	173,37 c€
U4	E1	81,98 c€	175,67 c€	U9	E1	94,70 c€	192,60 c€
	E2	23,49 c€	97,82 c€		E2	94,70 c€	192,60 c€
	E3	20,01 c€	93,18 c€		E3	92,72 c€	189,97 c€
U5	E1	68,92 c€	158,28 c€	U10	E1	81,27 c€	174,72 c€
	E2	68,92 c€	158,28 c€		E2	81,27 c€	174,72 c€
	E3	67,57 c€	156,49 c€		E3	79,80 c€	172,77 c€

Taula 7. Quadre resum dels resultats econòmics obtinguts en els diferents escenaris (Font: Propia)

S'ha de destacar que aquesta simulació s'ha dut a terme amb un conjunt reduït de dades referents a un dia 15/06, implicant així que les dades de consums, producció i els preus del mercat són del mateix dia. Aquesta reduïda selecció de dades és degut a que el procés d'una gran quantitat de dades amb el programa dissenyat es dificulta al treballar en una maquina virtual que simula el propi ordinador de treball. També, es necessari afegir que les dades del preu de mercat utilitzades són prèvies al creixement de preus en el mercat elèctric experimentat recentment i que a major diferència entre el preu de mercat i el preu de la venda d'excedents major es el benefici per a tots els usuaris de la comunitat. D'aquesta forma l'impacte de les variacions en el mercat elèctric es veu notablement mitigat.

A partir dels resultats econòmics, es denota una reducció del l'import respecte al terme d'energia entre l'escenari 3 i els escenaris 1 i 2. Aquesta reducció és lleugerament major en els casos dels "prosumers", o usuaris productors i consumidors, ja que acaben venent la gran majoria dels excedents generats a un major preu del que venien en l'escenari 2. La principal diferència entre l'escenari 2 i 3 i el perquè de la seva reducció es deu als 9172 Wh que es comercien de forma interna en la comunitat en l'escenari 3. En canvi en l'escenari 2 aquesta quantitat d'energia passa a ser abocada a la xarxa i venuda a preu d'excedents menor al preu que es comercialitza l'energia a la comunitat.

En el cas dels usuaris consumidors, la reducció es deu principalment a la compra d'energia a un preu menor al de mercat, el qual està vinculat a aquest i al preu dels excedents que ofereix la comercialitzadora. D'aquesta forma si el preu de mercat varia considerablement, al formar part de la comunitat, com a consumidor l'efecte seria menor ja que el preu d'aquests en els períodes amb producció solar a la comunitat seria menor que el de mercat.

Un dels principals avantatges de l'ús del software dissenyat per a la gestió d'una comunitat és l'ús de dades reals per al càlcul de l'energia generada individualitzada, del coeficient dinàmic de repartiment i del cost de l'energia. De manera que es calcula el coeficient dinàmic i en conseqüència l'energia generada en proporció amb els consums de cada usuari de la comunitat, evitant excedents de forma individual pels usuaris quant no n'hi ha pel conjunt de la comunitat. Per tant únicament en el cas en que la producció total de la comunitat sigui major als consums totals dels usuaris d'aquesta hi haurà excedents, i llavors directament es descomptarà el valor d'aquests als usuaris productors. També suposa l'ús dels preus hora real alhora de realitzar el còmput del terme d'energia. Així es descompta l'energia produïda i l'energia compartida de les hores amb un major cost d'energia del dia evitant càlculs de preu kWh mitjà mensual realitzats per les comercialitzadores amb que llavors descompten l'import total per energia produïda.

Seguint el comentat la figura dels usuaris productors s'afavoreix econòmicament per tal de potenciar l'entrada de més usuaris d'aquest tipus a la comunitat sense una major implicació per a la resta de consumidors associats a aquesta. D'altra banda, el fet d'utilitzar tecnologia "*blockchain*" per registrar les dades de cada usuari, permet que es publiquin de forma interna aquestes dades, oferint més transparència en el conjunt de la comunitat i augmentant la confiança dels usuaris en l'entitat. També suposa una reducció d'interaccions alhora de comercialitzar l'energia i es realitza sense discriminacions de cap tipus envers els membres de la comunitat, ja que els mètodes de càlcul són firmats per tota la comunitat. Un avantatge a sumar és la reducció de la carrega de consums que suporta el mercat elèctric al crear comunitats energètiques que utilitzin energia de producció renovable pròpia de forma autònoma.

Un dels desavantatges més importants actualment al fer ús d'una aplicació descentralitzada vinculada a la "*blockchain d'Ethereum*" és l'alt cost energètic que suposa el seu protocol de consens i les emissions que es vinculen a aquest consum.

Anàlisi de l'impacte ambiental

En l'aspecte ambiental, l'ús d'aquest programari suposa la reducció d'energia consumida de la xarxa per el conjunt de la comunitat. En el cas estudiat en l'escenari 3, on s'aplica el programari amb contractes intel·ligents desenvolupat, s'observa com respecte l'escenari 2 on no s'aplica el concepte de comunitat, el conjunt de la comunitat consumeix 9172 Wh menys de la xarxa elèctrica de transport i distribució (*ENGaut*).

Assumint el factor d'emissions segons els Valors aprovats en Comissió Permanent de Certificació Energètica de Edificis de 27 de Juny de 2013, l'electricitat del conjunt espanyol, l'hi corresponen $0,357 \text{ kg CO}_2 / \text{kWh } E.Final \cdot (62)$

$$E_f = E_{kwh} \cdot ENGaut \quad \text{Eq. 19}$$

Segons aquest valor i les dades aportades per la anàlisi realitzat, es dedueix a partir de l'Eq. 19 que el conjunt de la comunitat evita l'emissió de 3,274 kg CO₂ durant el període d'estudi. Suposant que es pogués extrapolar aquest valor de forma general per un any de funcionament de la comunitat energètica amb el programari desenvolupat, suposaria una reducció d'emissions per un valor total de 1195 t CO₂. S'especifica que a major participació en la comunitat tant d'usuaris productors com d'usuaris consumidors, com de la potència instal·lada d'energia renovables dels usuaris de la comunitat, aquest valors poden augmentar considerablement.

Tot i això, cal descomptar d'aquest còmput d'emissions evitades, les emeses per utilitzar la "blockchain d'"Ethereum". Actualment al treballar mitjançant l'ús del PoW, tot i que no s'han realitzat molts estudis sobre el tema i és necessària una major investigació, s'associa l'emissió de 46,54 kg de CO₂ per transacció realitzada. Partint d'aquests valor i comptabilitzant que es realitzen unes 240 transaccions al dia per la comunitat, s'estima que les emissions associades per l'ús de l'ecosistema "Ethereum" és de 4076 t CO₂. Tot i això, al llarg del 2022 és previsible un canvi de protocol de consens en la xarxa "Ethereum" passant al PoS. PoS consumeix una molt reduïda quantitat d'energia respecte el consum del PoW, concretament s'estima que el consum energètic seria 2000 vegades menor. D'aquesta forma, les emissions associades a l'ús d'aquesta tecnologia per al software que gestiona la comunitat energètica passen a reduir-se fins els 2,038 t CO₂. (63)

Al comptabilitzar les emissions associades a l'ús d'"Ethereum", segons els estudis i protocol de consens actual, s'observa com el còmput final acaba sent superior a les emissions evitades arribant als 2881 t CO₂. Tot i així, en el cas que finalment es dugui a terme aquest canvi de protocol de consens, les emissions finals es veurien considerablement afectades. Ja que el còmput total d'emissions seguiria evitant-ne més que no pas en n'emetria, amb un total de 1193 t CO₂.(64)



Conclusions

Resultats i Contribucions

Aquest treball te per fi crear una base de l'aplicació de la tecnologia "blockchain" en el sector energètic, concretament es centra en l'autoconsum d'energia renovable de forma compartida en una comunitat energètica. En la introducció del projecte, es repassa la situació global actual, concretament es comenta la implicació del consum d'energia elèctrica, el canvi climàtic en el planeta els últims anys, i les accions per mitigar l'efecte d'aquest canvi. Conjuntament es comenta breument la tecnologia "blockchain". A continuació, en el Capítol 3, es desenvolupa el concepte de comunitat energètica local i se'n exposa un exemple per a una millor comprensió. Seguidament s'ha realitzat una recerca dins el marc regulatori del sector elèctric espanyol on es busquen totes aquelles implicacions legals a que està associat l'autoconsum, i més concretament l'autoconsum col·lectiu. En el Capítol 5, es descriu la tecnologia "blockchain", de forma comprensible per tal de que es pugui captar els principals coneixements i aportacions d'aquesta. En el 5.1, es descriuen les principals plataformes per al desenvolupament d'aplicacions descentralitzades dins el panorama de la tecnologia "blockchain" actual. El capítol 6 planteja i desenvolupa el programari ideat per a la gestió de l'energia d'una comunitat energètica i la seva facturació mitjançant coeficients *ex post*. En el Capítol 7 es desenvolupa una simulació de 3 escenaris on aquest últim implica la utilització del programari dissenyat i desenvolupat en el Capítol 6 amb dades per un període de 24 h. Finalment, el capítol 8 exposa els beneficis ambientals de l'aplicació del programari creat durant el període d'estudi respecte l'escenari 2 proposat.

Aquest projecte contribueix tan en valor teòric com pràctic en diversos aspectes. Primer recopila els treballs més destacats que tracten el tema d'estudi. En segon terme, es mostren les implicacions legals dins el marc reglamentari espanyol de l'autoconsum i l'autoconsum col·lectiu, i la vinculació d'aquests amb les comunitats energètiques. En tercer lloc, es mostra el funcionament de la tecnologia "blockchain" i es fa ús d'una de les seves principals aplicacions desenvolupant les bases d'un programari per a la gestió energètica i econòmica d'una comunitat energètica amb autoconsum col·lectiu.

Aquest programari hi potencia la creació de comunitats energètiques sense la interacció d'una direcció central, essent autogestionades. També aporta un clar benefici tant econòmic, com ambiental i social, als usuaris de la comunitat a un mínim cost gràcies a l'ús de la tecnologia "blockchain". Tot i això, aquests beneficis necessiten d'una inversió prèvia per a la instal·lació de projectes que produeixen energia renovable dins la comunitat, i també es veuen vinculats al preu del mercat energètic i al de la

venda dels excedents establerts amb una comercialitzadora. Els resultats finals exposen un clar benefici econòmic tan pels participants productors com per els participants consumidors de la comunitat energètica. Aquest benefici són a canvi d'un cost mínim o nul segons les necessitats econòmiques del manteniment de la comunitat energètica i els quals poden augmentar considerablement segons les variacions en el mercat o els hàbits de consum en la comunitat. També s'aporten beneficis mediambientals els quals venen condicionats per l'ús de la tecnologia "blockchain" i protocol de consens que segueixi l'ecosistema.

Limitacions i treball futur

En el que respecte l'abast del projecte, s'ha d'afegir que al realitzar-lo mitjançant una tecnologia innovadora i degut a la limitada informació educativa que hi ha respecte la tecnologia "blockchain" el seu aprenentatge des de 0 és complex. Tot el realitzat en aquest treball ha estat a partir de l'autoaprenentatge i partint d'una base de partida nul·la en la tecnologia "blockchain". També destacar que aquesta tecnologia actualment encara és immadura i que necessita un major progrés en investigació. Tot i així s'ha trobat que encara no existeix un marc legal espanyol que vinculi l'ús d'aquesta tecnologia i les seves implicacions o restriccions legals en els aspectes d'interès d'aquest treball. També s'observa en la lectura científica, que la majoria d'implicacions entre el sector energètic i la comercialització d'energia P2P encara s'estan desenvolupant com a projectes pilot o estan en fases d'estudi en comunitats reduïdes. Al llarg dels anys aquesta situació pot evolucionar i el contingut d'aquest treball quedarà obsolet. També degut a les modificacions legals pertinents del govern espanyol, i la implicació que tinguin aquestes dins l'autoconsum col·lectiu poden determinar l'obsolescència d'aquest projecte.

Com a treball futur es proposa l'optimització del codi desenvolupat per al disseny del programari, i adaptar-lo a una forma flexible segons la tipologia de comunitat a la que es vulgui implementar. També en aquest aspecte, finalitzar la programació del conjunt de l'aplicació, referent als apartats externs a la tecnologia "blockchain", i estudiar la seva aplicació pràctica en una comunitat energètica real segons les premisses donades. Finalment, l'ús d'aquest programari desenvolupat en una "blockchain" privada per estudiar les implicacions del programari dins aquestes i poder realitzar un estudi de l'aplicació de comptadors intel·ligents com a nodes d'aquesta.

Conclusió final

L'evolució del sector elèctric degut a l'entrada de les tecnologies renovables, i el creixement d'aquestes degut a la conscienciació sobre l'escalfament del planeta, està avançant ràpidament. A més, l'augment del consum elèctric en contra de l'ús d'altres recursos energètics més contaminants, posa més pes en

el sistema elèctric actual. Aquesta consciència social sobre l'escalfament global, potencia la col·laboració ciutadana per a la solució de problemes de caràcter mediambiental o energètic i d'implicació econòmica per aquests. Conjuntament a aquests fets, ha nascut la tecnologia "*blockchain*", aportant independència de tercers, transparència i seguretat per un reduït cost, la qual es aplicable en molts sectors, tot i que s'ha d'estudiar la seva implicació.

Aquest treball exposa els principals progressos en l'àmbit del comerç d'energia P2P. També es comparen els principals ecosistemes que permeten el desenvolupament d'aplicacions descentralitzades i les característiques d'aquests. Finalment es descriu el disseny i programació del codi per uns "*smart-contracts*" que assenten les bases a una aplicació descentralitzada per al comerç d'energia dins una comunitat energètica local. Aquesta aplicació registra les dades aportades per comptadors intel·ligents i a partir d'aquestes realitza el càlcul de l'import total de facturació que finalment l'usuari ha de firmar per realitzar el pagament. Aquest càlcul s'estipula tenint present que el usuaris formen part d'una comunitat energètica local, fent ús de l'autoconsum col·lectiu, on els propis usuaris poden ser estipulats com a productors d'energia. Definitivament aporta un benefici econòmic ambiental i social envers a tots els seus usuaris, independentment de la seva tipologia, potenciant el pas de consumidors no interessats en aquest aspecte cap a l'autoconsum. Això facilita el progrés de la transició energètica i apropa el col·lectiu a una economia descarbonitzada i independent energèticament.

Pressupost i/o Anàlisi Econòmica

Per l'anàlisi econòmica a realitzar, s'ha considerat un salari mensual de 1200€, conjuntament amb les càrregues socials associades a aquest, les quals fan ascendir aquest concepte fins uns 1680 € mensuals. Si a més a més, es té en compte la remuneració de vacances i pagues extres, el total mensual ascendeix fins els 1980€. Conjuntament amb el cost dels components necessaris per al desenvolupament d'aquest també es té en compte un 15 % adicional corresponent al benefici industrial. Finalment la suma dels conceptes anteriors conforma la base imposable i se li aplica un 21 % per acabar de definir l'import total. En la Taula 8 es concreten els apartats i especifica els imports de cada un dels aspectes del pressupost.

Referència	Descripció	Nombre	Preu Unitari	Preu Total
S.01	Sou enginyer en formació, segons conveni de treball i càrregues socials associades	3 mesos	1980 €/mes	5940,00 €
P.01	Portàtil Asus R560UD-EJ387 + Microsoft Windows 10 Home 64Bits OEM	1 u.	866,95 €/u.	866,95 €
O.01	Llicència Microsoft Office	1 u.	149 €/u.	149,00 €
B.01	Benefici Industrial		15%	1043,40 €
Base Imposable				7999,34 €
I.01	IVA		21 %	1679,86 €
Total				9679,20 €

Taula 8. Pressupost de la inversió necessària pel projecte (Font: Propia)

Pressupost vàlid per a 6 mesos.

Bibliografia

1. Introduction to Smart Contracts — Solidity 0.8.10 documentation. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://docs.soliditylang.org/en/v0.8.10/introduction-to-smart-contracts.html>.
2. Solidity - Quick Guide. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: https://www.tutorialspoint.com/solidity/solidity_quick_guide.htm.
3. El Glosario de los Smart Contracts de Nick Szabo. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://academy.bit2me.com/glosario-smart-contract-nick-szabo/>.
4. Institut d'Estudis Catalans - Diec2. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://dlc.iec.cat/Results?DecEntradaText=consorci&AllInfoMorf=False&OperEntrada=0&OperDef=0&OperEx=0&OperSubEntrada=0&OperAreaTematica=0&InfoMorfType=0&OperCatGram=False&AccentSen=False&CurrentPage=0&refineSearch=0&Actualitzacions=False>.
5. Glasgow Leaders' Declaration on Forests and Land Use - UN Climate Change Conference (COP26) at the SEC – Glasgow 2021. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://ukcop26.org/glasgow-leaders-declaration-on-forests-and-land-use/>.
6. Blockchain y Energías Renovables - YouTube. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=FyIWBOLTSvc&ab_channel=AlbertoRosandi.
7. Centralizado Descentralizado y Distribuido, de la Pirámide al Círculo. | Bienestar Mutuo. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://bienestarmutuo.org/centralizado-descentralizado-distribuido-lo-horizontal-a-vertical/>.
8. Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. A: . 2009, Vol. 0, núm. 0, p. 9. ISSN 17506166. DOI 10.1108/TG-06-2020-0114.
9. How does a blockchain work - Simply Explained - YouTube. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=SSo_ElwHSd4&ab_channel=SimplyExplained.
10. Freitag, F. et al. A look at energy efficient system opportunities with community network clouds. A: *CEUR Workshop Proceedings*. 2014, Vol. 1203, p. 7-9. ISSN 16130073.
11. Yang, Q. et al. Blockchain-based decentralized energy management platform for residential distributed energy resources in a virtual power plant. A: *Applied Energy* [en línia]. Elsevier Ltd, 2021, Vol. 294, núm. March, p. 117026. ISSN 03062619. DOI 10.1016/j.apenergy.2021.117026. Disponible a: <https://doi.org/10.1016/j.apenergy.2021.117026>.
12. Lin, J., Pipattanasomporn, M. i Rahman, S. Comparative analysis of auction mechanisms and bidding strategies for P2P solar transactive energy markets. A: *Applied Energy*. 2019, Vol. 255, núm. October. ISSN 03062619. DOI 10.1016/j.apenergy.2019.113687.
13. Sreeramulu Mahesh, G. et al. Energy management with blockchain technology in DC microgrids. A: *Materials Today: Proceedings* [en línia]. Elsevier Ltd, 2021, Vol. 47, núm. xxxx, p. 2232-2236. ISSN 22147853. DOI 10.1016/j.matpr.2021.04.188. Disponible a:

<https://doi.org/10.1016/j.matpr.2021.04.188>.

14. Simaremare, A.A. et al. Suitability study of Blockchain application in electric utility company business processes. A: *IOP Conference Series: Materials Science and Engineering*. 2021, Vol. 1098, núm. 5, p. 052105. ISSN 1757-8981. DOI 10.1088/1757-898x/1098/5/052105.

15. Zhu, S. et al. The development of energy blockchain and its implications for China's energy sector. A: *Resources Policy*. 2020, Vol. 66, núm. December 2019. ISSN 03014207. DOI 10.1016/j.resourpol.2020.101595.

16. O'Donovan, P. i O'Sullivan, D.T.J. A systematic analysis of real-world energy blockchain initiatives. A: *Future Internet*. 2019, Vol. 11, núm. 8. ISSN 19995903. DOI 10.3390/fi11080174.

17. Galici, M. et al. Energy blockchain for public energy communities. A: *Applied Sciences (Switzerland)*. 2021, Vol. 11, núm. 8, p. 1-15. ISSN 20763417. DOI 10.3390/app11083457.

18. Wang, L. et al. Adaptive bidding strategy for real-time energy management in multi-energy market enhanced by blockchain. A: *Applied Energy*. 2020, Vol. 279, núm. August. ISSN 03062619. DOI 10.1016/j.apenergy.2020.115866.

19. Van Leeuwen, G. et al. An integrated blockchain-based energy management platform with bilateral trading for microgrid communities. A: *Applied Energy*. 2020, Vol. 263, núm. October 2019. ISSN 03062619. DOI 10.1016/j.apenergy.2020.114613.

20. Wongthongtham, P. et al. Blockchain-enabled Peer-to-Peer energy trading. A: *Computers and Electrical Engineering* [en línia]. Elsevier Ltd, 2021, Vol. 94, núm. April, p. 107299. ISSN 00457906. DOI 10.1016/j.compeleceng.2021.107299. Disponible a: <https://doi.org/10.1016/j.compeleceng.2021.107299>.






21. Zhang, C., Yang, T. i Wang, Y. Peer-to-Peer energy trading in a microgrid based on iterative double auction and blockchain. A: *Sustainable Energy, Grids and Networks* [en línia]. Elsevier Ltd, 2021, Vol. 27, p. 100524. ISSN 23524677. DOI 10.1016/j.segan.2021.100524. Disponible a: <https://doi.org/10.1016/j.segan.2021.100524>.

22. Schlund, J., Ammon, L. i German, R. ETHome: Open-source blockchain based energy community controller. A: *e-Energy 2018 - Proceedings of the 9th ACM International Conference on Future Energy Systems*. 2018, p. 319-323. DOI 10.1145/3208903.3208929.

23. Petri, I. et al. Blockchain for energy sharing and trading in distributed prosumer communities. A: *Computers in Industry* [en línia]. Elsevier B.V., 2020, Vol. 123, p. 103282. ISSN 01663615. DOI 10.1016/j.compind.2020.103282. Disponible a: <https://doi.org/10.1016/j.compind.2020.103282>.

24. Van Cutsem, O. et al. Cooperative energy management of a community of smart-buildings: A Blockchain approach. A: *International Journal of Electrical Power and Energy Systems* [en línia]. Elsevier, 2020, Vol. 117, núm. October 2019, p. 105643. ISSN 01420615. DOI 10.1016/j.ijepes.2019.105643. Disponible a: <https://doi.org/10.1016/j.ijepes.2019.105643>.

25. Afzal, M. et al. Blockchain enabled distributed demand side management in community energy system with smart homes. A: *IEEE Access*. IEEE, 2020, Vol. 8, p. 37428-37439. ISSN 21693536. DOI 10.1109/ACCESS.2020.2975233.

-
26. Wang, L. et al. Credible peer-to-peer trading with double-layer energy blockchain network in distributed electricity markets. A: *Electronics (Switzerland)*. 2021, Vol. 10, núm. 15, p. 1-26. ISSN 20799292. DOI 10.3390/electronics10151815.
27. Nidhin Mahesh, A., Sai Shibu, N.B. i Balamurugan, S. Conceptualizing Blockchain based Energy Market for Self Sustainable Community. A: *BlockSys 2019 - Proceedings of the 2019 Workshop on Blockchain-Enabled Networked Sensor Systems*. 2019, p. 1-7. DOI 10.1145/3362744.3363345.
28. Karandikar, N., Chakravorty, A. i Rong, C. Blockchain based transaction system with fungible and non-fungible tokens for a community-based energy infrastructure. A: *Sensors*. 2021, Vol. 21, núm. 11. ISSN 14248220. DOI 10.3390/s21113822.
29. Saxena, S. et al. Design and Field Implementation of Blockchain Based Renewable Energy Trading in Residential Communities. A: *2nd International Conference on Smart Grid and Renewable Energy, SGRE 2019 - Proceedings*. 2019, DOI 10.1109/SGRE46976.2019.9020672.
30. Remix - Ethereum IDE. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: <https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.7.6+commit.7338295f.js>.
31. Parlamento Europeo y Consejo Europeo. Directiva (UE) 2018/2001 relativa al fomento del uso de energía procedente de fuentes renovables. A: *Diario Oficial de la Unión Europea*. 2018, Vol. 2018, núm. 87, p. 141-144. ISSN 1989-5666.
32. Parlamento Europeo y Consejo Europeo. Directiva (Ue) 2019/944 Del Parlamento Europeo Y Del Consejo. A: *Boletín Oficial del Estado*. 2019, Vol. 2019, núm. 2, p. 210-230.
33. Institute for Energy Diversification and Saving - IDAE. Guía para el Desarrollo de Instrumentos de Fomento de Comunidades Energéticas Locales. A: *Journal of Chemical Information and Modeling*. 2013, Vol. 53, núm. 9, p. 1689-1699. ISSN 1098-6596.
34. Diputació de Barcelona. Guía para el impulso de comunidades energéticas con perspectiva municipal. A: [en línia]. 2021, Disponible a: www.dauss.es.
35. Comunitats energètiques: projectes transformadors!      - YouTube. A: [en línia]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=mrWQ3yzG2GU&ab_channel=Ecoserveis.
36. JEFATURA DEL ESTADO. Ley 24/2013, de 26 de diciembre, del Sector Eléctrico. A: *Boletín Oficial del Estado*. 2009, p. 61561-61567.
37. Jefatura del Estado. Real Decreto-ley 15/2018, de 5 de octubre, de medidas urgentes para la transición energética y la protección de los consumidores. A: *Boe*. 2018, Vol. 6 Octubre, núm. 242, p. 97430-97467.
38. MINISTERIO PARA LA TRANSICIÓN ECOLÓGICA. Real Decreto 244/2019, de 5 de abril, por el que se regulan las condiciones administrativas, técnicas y económicas del autoconsumo de energía eléctrica. A: *Boletín Oficial del Estado*. 2009, p. 61561-61567.
39. MINISTERIO PARA LA TRANSICIÓN ECOLÓGICA Y EL RETO DEMOGRÁFICO. Orden TED/1247/2021,

de 15 de noviembre. A: *Boletín Oficial del Estado*. 2021, p. 26798-26800.

40. ¿Qué es el BLOCKCHAIN? Explicado por un INGENIERO INFORMÁTICO - (Bitcoin, NFTs y más) - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=V9Kr2SujqHw&ab_channel=DotCSV.

41. Entiende Bitcoin y Ethereum - Explicación técnica a fondo en español sobre Criptomonedas - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=YBNr69vrscw&list=PL-uuYYusfTS0gmFLZDpo0K7w4yYGM08S-&index=3&ab_channel=NateGentile.

42. Comparativa de los mejores clientes BitTorrent para Linux | Linux Adictos. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://www.linuxadictos.com/todo-sobre-los-mejores-clientes-bittorrent-para-linux.html>.

43. Protocolos de consenso - Blockchain Federal Argentina. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://bfa.ar/blockchain/protocolos-de-consenso>.

44. Top 6 smart contract platforms: a deep dive - Itransition. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://www.itransition.com/blog/smart-contract-platforms>.

45. ¿Qué es Polkadot (DOT)? A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://academy.bit2me.com/que-es-polkadot-dot/>.

46. Nominated Proof-of-Stake — Research at W3F. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://research.web3.foundation/en/latest/polkadot/NPoS/index.html>.

47. Hyperledger Fabric - ¿Qué es Hyperledger? en Español | by Victor Muñoz Reyes | Babel — go2chain | Medium. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://medium.com/babel-go2chain/hyperledger-fabric-qué-es-hyperledger-en-español-d13f8d144455>.

48. ¿Qué es Tezos XTZ y qué características presenta su funcionamiento? A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://economia3.com/que-es-tezos-xtz-criptomoneda/>.

49. ¿Qué es Stellar (XLM)? A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://academy.bit2me.com/que-es-stellar-xml-lumens-criptomoneda/>.

50. Solana: Qué es y cómo funciona su novedosa blockchain. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://www.profesionalreview.com/2021/07/10/que-es-solana/>.

51. Intro to ether | ethereum.org. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: <https://ethereum.org/en/developers/docs/intro-to-ether/>.

52. 😎 Solidity Tutorial - Introducción a los Smart Contracts en Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=iwMMHDilspg&ab_channel=AlbertoLasa-Blockchain.

53. 😎 Solidity Tutorial - Primer contrato - Smart Contracts en Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=M6Arz3wGKTU&ab_channel=AlbertoLasa-Blockchain.

-
54. Como programar en Ethereum en 2021 (1/2) | Curso de Solidity para principiantes - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=Dm4oRsxQKbE&ab_channel=EscuelaCryptoES.
55. Types — Solidity 0.8.10 documentation. A: [en línea]. [Consulta: 16 gener 2022]. Disponible a: <https://docs.soliditylang.org/en/v0.8.10/types.html>.
56. 😎 Solidity Tutorial - Transacciones y Mensajes - Smart Contracts en Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=q_8CuhE9II&t=644s&ab_channel=AlbertoLasa-Blockchain.
57. 😎 Solidity Tutorial - Transacciones - Smart Contracts en Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=G9Sgwez1U9I&t=1063s&ab_channel=AlbertoLasa-Blockchain.
58. 😎 Solidity Tutorial - Array, Mapping y Struct - Smart Contracts en Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=_PbU9TE_7bY&t=287s&ab_channel=AlbertoLasa-Blockchain.
59. 😎 Solidity Tutorial - Iteración de listas con mapping - Ethereum - YouTube. A: [en línea]. [Consulta: 12 gener 2022]. Disponible a: https://www.youtube.com/watch?v=9JcZ2EUTkfk&t=1083s&ab_channel=AlbertoLasa-Blockchain.
60. Limia, E.G. et al. Blockchain Technology Application to Energy Communities. A: [en línea]. 2018, núm. July. Disponible a: <https://repositorio.comillas.edu/xmlui/handle/11531/21565>.
61. Day-ahead hourly price | OMIE. A: [en línea]. [Consulta: 16 gener 2022]. Disponible a: <https://www.omie.es/en/market-results/daily/daily-market/daily-hourly-price>.
62. Ministerio de Industria, E. y T. Factores de emisión de CO2 y coeficientes de paso a energía primaria de diferentes fuentes de energía final consumidas en el sector de edificios en España. A: *Documento Reconocido del Reglamento de Instalaciones Térmicas en los Edificios (RITE)* [en línea]. 2016, p. 16, 17, 18. ISSN 1098-6596. Disponible a: http://www.minetad.gob.es/energia/desarrollo/EficienciaEnergetica/RITE/Reconocidos/Reconocidos/Otros documentos/Factores_emision_CO2.pdf.
63. Cuánto coste energético tiene realmente publicar y comercializar un NFT: hablemos de Ethereum 2.0 y camisetas de algodón. A: [en línea]. [Consulta: 16 gener 2022]. Disponible a: <https://www.xataka.com/criptomonedas/cuanto-coste-energetico-tiene-realmente-publicar-comercializar-nft-hablemos-ethereum-2-0-camisetas-algodon>.
64. Ethereum consumiría 99% menos energía con la prueba de participación. A: [en línea]. [Consulta: 16 gener 2022]. Disponible a: <https://www.criptonoticias.com/tecnologia/ethereum-consumiria-99-menos-energia-prueba-participacion/>.



Annex A

A1. Codi Registre Usuaris

```
2 //SPDX-License-Identifier: MIT
3 pragma solidity >=0.5.0;
4 contract Registreusuaris {
5     address owner;
6     uint256 preu_servei;
7     uint256 identificadors_contadors;
8     mapping(uint256 => SmartM ) public contadors;
9     struct SmartM {
10         uint256 Identificador;
11         string TipusI;
12         uint256 DNI_usuari;
13         address U_address;
14         uint32 kWpi;
15         uint32 kWpc;
16     }
17
18     modifier preciofiltro(uint256 _precio) {
19         require(_precio == preu_servei);
20         _;
21     }
22
23     modifier onlyOwner {
24         require(msg.sender == owner);
25         _;
26     }
27
28     constructor (uint256 _precio) {
29         owner = msg.sender;
30         preu_servei = _precio;
31     }
32
```

```
33  function addSmartM( uint256 _DNI_usuari, uint32 _kWpi, uint32 _kWpc, string memory
    _instalacio_tipus) public preciofiltro(msg.value) payable {
34      contadors[identificadors_contadors] = SmartM(identificadors_contadors, _instalacio_tipus,
    _DNI_usuari, msg.sender, _kWpi, _kWpc);
35      identificadors_contadors++;
36  }
37
38  function getidentificadors_contadors() external view returns(uint256) {
39      return (identificadors_contadors);
40  }
41
42  function getAllSmartM() public view onlyOwner returns(uint256[] memory, uint256[] memory,
    address[] memory, uint256[] memory, uint256[] memory){
43      uint256[] memory identificador = new uint256[](identificadors_contadors);
44      uint256[] memory dni_usuari = new uint256[](identificadors_contadors);
45      address[] memory u_address = new address[](identificadors_contadors);
46      uint256[] memory kwpi = new uint256[](identificadors_contadors);
47      uint256[] memory kwpc = new uint256[](identificadors_contadors);
48      for (uint256 i = 0; i < identificadors_contadors; i++) {
49          identificador[i] = contadors[i].Identificador;
50          dni_usuari[i] = contadors[i].DNI_usuari;
51          u_address[i] = contadors[i].U_address;
52          kwpi[i] = contadors[i].kWpi;
53          kwpc[i] = contadors[i].kWpc;
54      }
55      return ( identificador, dni_usuari, u_address, kwpi, kwpc);
56  }
57 }
```

A2. Codi alfaList

```
1 //SPDX-License-Identifier: MIT
2 pragma solidity >=0.7.6;
3
4 contract alfaList {
5     mapping(uint256=> BE) public Registre;
6     struct BE {
7         uint256 ID_user;
8         uint256 Time;
9         uint256 EpWh;
10        uint256 EcWh;
11        uint256 Alfa;
12        uint256 Excedents;
13    }
14    uint256 addressRegistryCount;
15
16
17    function set(uint256 _ID_user, uint256 _Time, uint256 _EpWh, uint256 _EcWh) public {
18        Registre[addressRegistryCount] = BE( _ID_user, _Time, _EpWh, _EcWh, 100*_EpWh/_EcWh,
19        _EpWh - _EcWh);
20        addressRegistryCount++;
21    }
22
23    mapping(uint256 => uint256) public PreuMM;
24    function setPreu(uint256 _temps, uint256 _PreuMM)public{
25        PreuMM[_temps] = _PreuMM;
26    }
27
28    function getAll() public view returns ( uint256[] memory, uint256[] memory, uint256[] memory,
29    uint256[] memory, uint256[] memory ){
30        uint256[] memory id_user = new uint256[](addressRegistryCount);
31        uint256[] memory time = new uint256[](addressRegistryCount);
32        uint256[] memory epWh = new uint256[](addressRegistryCount);
33        uint256[] memory ecWh = new uint256[](addressRegistryCount);
34        uint256[] memory alfa = new uint256[](addressRegistryCount);
35        uint256[] memory excedents = new uint256[](addressRegistryCount);
```

```

34     for (uint256 i = 0; i < addressRegistryCount; i++) {
35         id_user[i] = Registre[i].ID_user;
36         time[i] = Registre[i].Time;
37         epWh[i] = Registre[i].EpWh;
38         ecWh[i] = Registre[i].EcWh;
39         alfa[i] = Registre[i].Alfa;
40         excedents[i] = Registre[i].Excedents;
41     }
42     return ( id_user, time, epWh, ecWh, alfa);
43 }
44
45 mapping(uint256 => DadesT) public DadesH;
46 struct DadesT {
47     uint256 amountET;//Quantitat d'excedents en l'hora h
48     uint256 amountUE;// Quantitat d'usuaris amb excedents alfa >= 1
49     uint256 amountUA;// Quantitat d'usuaris productors sense excedents 0 <=alfa < 1
50     uint256 amountUC;// Quantitat d'usuaris unicament consumidors alfa = 0
51     uint256 ENGh;// Quantitat d'energia generada en l'hora h
52     uint256 ENCh;// Quantitat d'energia consumida en l'hora h
53     uint256 ENCPH;// Quantitat d'energia consumida parcial en l'hora h =>
54     //correspont a l'energia consumida dels UC més la diferencia de l'energia consumida i
l'energia produïda dels productors sense excedents
55 }
56
57 function setDadesT() public {
58     for (uint256 t = 0; t <= Registre[addressRegistryCount - 1].Time; t++ ) {
59
60         // Declaració de variables internes per a la posterior introducció a DadesH
61
62         uint256 _amountET;
63         uint256 _amountUE;
64         uint256 _amountUA;
65         uint256 _amountUC;
66         uint256 _ENGh;
67         uint256 _ENCh;
68         uint256 _ENCPH;
69

```

```

70     for (uint i = 0; i < addressRegistryCount; i++){
71         if ( t == Registre[i].Time){
72
73             if (Registre[i].Alfa >= 100 ){
74                 _amountET += Registre[i].Excedents;
75                 _amountUE++;
76             }
77             if (0 < Registre[i].Alfa && Registre[i].Alfa < 100 ){
78                 _amountUA++;
79                 _ENCPh += Registre[i].EcWh - Registre[i].EpWh;
80             }
81             if (Registre[i].Alfa == 0 ){
82                 _amountUC++;
83                 _ENCPh += Registre[i].EcWh;
84             }
85             _ENGh += Registre[i].EpWh;
86             _ENCh += Registre[i].EcWh;
87         }
88     }
89
90     DadesH[t] = DadesT( _amountET, _amountUE, _amountUA, _amountUC, _ENGh, _ENCh,
    _ENCPh);
91 }
92 }
93
94 mapping( uint256 => mapping( uint256 => DadesHI_ ) ) public DadesHI;// primer uint256 es
    l'ID_user
95 //segon uint256 correspon a l'hora per al conjunt de dades horaries individualitzades
96 struct DadesHI_{
97     uint256 ENPhi; //Energia produïda horaria individualitzada
98     uint256 ENGhi; //Energia generada horaria individualitzada
99     uint256 ENChi; //Energia consumida horaria individualitzada
100    uint256 ENCrhi; //Energia consumida restant horaria individualitzada (només UA)
101    uint256 ENCRhi; //Energia consumida de xarxa horaria individualitzada
102    uint256 Eautci; //Energia autoconsumida horaria compartida
103    uint256 Eauti; //Energia autoconsumida horaria individualitzada
104    uint256 Betahi; //Coeficient de repartiment horari individualitzat

```

```
105     uint256 PreuMMh;//Preu mercat majorista en la corresponent hora
106 }
107
108
109 //Variable que conti la quantitat de períodes t d'una hora que tinguis per llavors limitar el càlcul
    dels descomptes en les factures mensuals.
110
111 function setBeta() public {
112     for(uint256 i = 0; i < addressRegistryCount; i++){
113
114         // Declaració de variables internes per a la posterior introducció a DadesHI
115
116         uint256 _ENPhi = Registre[i].EpWh;
117         uint256 _ENGhi;
118         uint256 _ENChi = Registre[i].EcWh;
119         uint256 _ENCrhi;
120         uint256 _ENCRhi;
121         uint256 _Eautci;
122         uint256 _Eauti;
123         uint256 _Betahi;
124         uint256 _PreuMMh = PreuMM[Registre[i].Time];
125
126         if (Registre[i].Alfa >= 100 ){
127
128             _ENCrhi = 0;
129             _Betahi = (_ENChi * 1000000) / DadesH[Registre[i].Time].ENGh ;
130             _ENGhi = (_Betahi * DadesH[Registre[i].Time].ENGh) / 1000000 ;
131             _Eautci = 0;
132             _Eauti = _ENGhi;
133             _ENCRhi = 0;
134
135         }
136         if (0 <= Registre[i].Alfa && Registre[i].Alfa < 100 ){
137
138             _ENCrhi = _ENChi - _ENPhi;
```



```

139     _Betahi = (_ENPhi * 1000000) / DadesH[Registre[i].Time].ENGh + (_ENCrhi *
DadesH[Registre[i].Time].amountET * 1000000) / (DadesH[Registre[i].Time].ENCPh *
DadesH[Registre[i].Time].ENGh);
140     _ENGhi = (DadesH[Registre[i].Time].ENGh * _Betahi) / 1000000;
141     if(_ENGhi >= _ENChi){
142         _Eauti = _ENChi;
143     } else {
144         _Eauti = _ENGhi;
145     }
146     _Eautci = _Eauti - _ENPhi;
147     _ENCrhi = _ENChi - _Eauti;
148
149 }
150 if (Registre[i].Alfa == 0){
151
152     _ENCrhi = 0;
153     _Betahi = (_ENChi * DadesH[Registre[i].Time].amountET * 1000000) /
(DadesH[Registre[i].Time].ENCPh * DadesH[Registre[i].Time].ENGh);
154     _ENGhi = (_Betahi * DadesH[Registre[i].Time].ENGh) / 1000000;
155     if(_ENGhi >= _ENChi){
156         _Eauti = _ENChi;
157     } else {
158         _Eauti = _ENGhi;
159     }
160     _Eautci = _Eauti;
161     _ENCrhi = _ENChi - _Eauti;
162
163 }
164
165     DadesHI[Registre[i].ID_user][Registre[i].Time] = DadesHI_( _ENPhi, _ENGhi, _ENChi,
    _ENCrhi, _ENCrhi, _Eautci, _Eauti, _Betahi, _PreuMMh);
166 }
167 }
168
169 function getDadeshi() private view returns ( uint256[] memory, uint256[] memory, uint256[]
memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[] memory, uint256[]
memory, uint256[] memory){

```

```
170     uint256[] memory enphi = new uint256[](addressRegistryCount);
171     uint256[] memory enghi = new uint256[](addressRegistryCount);
172     uint256[] memory enchi = new uint256[](addressRegistryCount);
173     uint256[] memory encrhi = new uint256[](addressRegistryCount);
174     uint256[] memory encRhi = new uint256[](addressRegistryCount);
175     uint256[] memory eautci = new uint256[](addressRegistryCount);
176     uint256[] memory eauti = new uint256[](addressRegistryCount);
177     uint256[] memory betta = new uint256[](addressRegistryCount);
178     uint256[] memory preuMMh = new uint256[](addressRegistryCount);
179     for(uint256 i = 0; i < addressRegistryCount; i++){
180         enphi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENPhi;
181         enghi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENGhi;
182         enchi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENChi;
183         encrhi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENCRhi;
184         encRhi[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].ENCRhi;
185         eautci[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Eautci;
186         eauti[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Eauti;
187         betta[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].Betahi;
188         preuMMh[i] = DadesHI[Registre[i].ID_user][Registre[i].Time].PreuMMh;
189     }
190     return (enphi, enghi, enchi, encrhi, encRhi, eautci, eauti, betta, preuMMh);
191 }
192 }
193 }
194
```

A3. Codi Facturació

```
1 //SPDX-License-Identifier: MIT
2 pragma solidity >=0.5.0;
3
4 contract Facturacio{
5
6     address public alfa_List;
7     address public Registre_usuaris;
8
9     constructor(address _addressC, address _addressD) public {
10
11         alfa_List = _addressC;
12         Registre_usuaris = _addressD;
13     }
14
15     mapping(uint256 => mapping (uint256 => Factura)) public Factures;//1ª Clau DNI_usuari, 2ª
        clau Identificador Factura
16     struct Factura { //Totes les unitats monetaries estaran en cèntims d'Euro.
17         uint256 Identificador;
18         uint256 DNI_usuari;
19         address U_address;
20         uint32 kWc;
21         uint256 numfact;
22         uint256 kWhcr;
23         uint256 kWhautc;
24         uint256 kWhautT;
25         uint256 kWhCT;
26         uint256 kWhe;
27         uint256 PreuKWc;//Cost potència contractada.
28         uint256 PreukWhautc;//Preu promig de l'energia autoconsumida compartida.
29         uint256 PreukWhe;//Preu promig de l'energia excedentaria.
30         uint256 PreukWhcr;// Promig total energia consumida xarxa.
31         uint256 cEurPC;
32         uint256 cEurEU;
33         uint256 cEurIE;//5.113% del terme de potència i Energia consumida total.
34         uint256 cEurOther;//Terme per energia compartida a la comunitat.
```

```
35     uint256 cEurIVA;//IVA 10%.
36     uint256 cEurAll;
37 }
38
39 function setfactures() public{
40
41     (bool success, bytes memory Rcount) =
42     Registre_usuaris.call(abi.encodeWithSignature('identificadors_contadors'));
43     require(success, "Error");
44     (uint256 _idcount) = abi.decode(Rcount, (uint256));
45     for(uint256 i = 0; i < _idcount; i++){
46         (bool success1, bytes memory data) =
47         Registre_usuaris.call(abi.encodeWithSignature('contadors[uint256]',i));
48         require(success1, "Error");
49         (uint256 _id, string memory _tipusi, uint256 _DNI_usuari, address _U_Address, uint32
50         _kWpi, uint32 _kWpc) = abi.decode(data, (uint256, string, uint256, address, uint32, uint32));
51         uint256 _numfact;
52         uint256 _kWhcr;
53         uint256 _kWhautc;
54         uint256 _KWhautT;
55         uint256 _kWhCT;
56         uint256 _kWhc;
57         uint256 _PreuKWc;
58         uint256 _PreukWhautc;
59         uint256 _PreukWhe;
60         uint256 _PreukWhcr;
61         uint256 _cEurPC;
62         uint256 _cEurEU;
63         uint256 _cEurIE;
64         uint256 _cEurOther;
65         uint256 _cEurIVA;
66         uint256 _cEurAll;
67         uint256 _cEurautci;
68         uint256 _cEurR;
69         uint256 Costkwceh;
```

```

69     for(uint256 t = 0; t < 24; t++){
70         (bool success2, bytes memory data1) =
            alfa_List.call(abi.encodeWithSignature('DadesHI[uint256][uint256]', _DNI_usuari, t));
71         require(success2, "Error");
72         (uint256 _ENPhi, uint256 _ENGhi, uint256 _ENChi, uint256 _ENCrhi, uint256 _ENCRhi,
            uint256 _Eautci, uint256 _Eauti, uint256 _Betahi, uint256 _PreuMMh) = abi.decode(data1,
            (uint256, uint256, uint256, uint256, uint256, uint256, uint256, uint256, uint256));
73
74         _kWhcr += _ENCRhi;
75         _kWhautc += _Eautci;
76         _KWhautT += _Eauti;
77         _kWhCT += _ENChi;
78         Costkwceh = (7 + _PreuMMh)/2;
79         if(_ENPhi > _ENChi){
80             _kWhe += _ENPhi - _ENChi;
81             _cEurOther -= Costkwceh*( _ENPhi - _ENChi);
82         }
83         else{
84             _kWhe += 0;
85             _cEurOther += 0;
86         }
87         _cEurEU += Costkwceh*_Eautci + _PreuMMh*_ENCRhi;
88         _cEurautci += Costkwceh*_Eautci;
89         _cEurR += _PreuMMh*_ENCRhi;
90     }
91
92     _PreukWhautc = _cEurautci/_kWhautc;
93     _PreukWhcr= _cEurR/_kWhcr;
94     _PreukWhe = _cEurOther/_kWhe;
95     _cEurPC = _PreukWc * _kWpc;
96     _cEurIE = ((_cEurEU + _cEurPC) * 5113)/100000;
97     _cEurIVA = ((_cEurEU + _cEurPC + _cEurIE + _cEurOther)*10)/100;
98     _cEurAll = _cEurEU + _cEurPC + _cEurIE + _cEurOther + _cEurIVA;
99
100    Factures[_DNI_usuari][_numfact] =Factura( _id, _DNI_usuari, _U_Address, _kWpc,
        _numfact, _kWhcr, _kWhautc, _KWhautT, _kWhCT, _kWhe, _PreukWc,

```

```
101     _PreukWhautc, _PreukWhe, _PreukWhcr, _cEurPC, _cEurEU, _cEurIE, _cEurOther,  
102     _cEurIVA, _cEurAll);  
103 }  
104  
105 function PayFact(uint256 _DNI_usuari, uint256 _numfact) public payable{  
106     require(msg.value == Factures[_DNI_usuari][_numfact].cEurAll*1000000);  
107 }  
108 }
```

A4. Dades Escenari 1

S1 (Wh)	H	S2 (Wh)	H	S3 (Wh)	H	S4 (Wh)	H	S5 (Wh)	H	S6 (Wh)	H
652,7	1	238,6	1	266,3	1	184,7	1	189,5	1	342,8	1
436,9	2	298,8	2	267,6	2	715,2	2	714,8	2	210,7	2
173,4	3	191,6	3	174,6	3	524,3	3	183,1	3	133,4	3
227,9	4	286,1	4	178,9	4	156,6	4	228,8	4	142,0	4
154,4	5	181,3	5	169,0	5	227,5	5	160,4	5	210,7	5
191,0	6	180,2	6	175,9	6	193,9	6	132,2	6	152,7	6
277,0	7	162,5	7	140,2	7	178,5	7	133,9	7	247,6	7
1100,9	8	373,8	8	741,1	8	526,8	8	237,5	8	432,1	8
461,1	9	260,4	9	377,3	9	182,0	9	386,0	9	208,3	9
1044,7	10	163,1	10	346,3	10	125,5	10	190,2	10	272,1	10
374,8	11	818,1	11	448,5	11	230,9	11	774,2	11	788,4	11
560,2	12	224,5	12	155,0	12	312,2	12	261,9	12	346,5	12
166,4	13	203,0	13	115,2	13	168,9	13	310,7	13	272,5	13
212,2	14	218,1	14	262,8	14	115,2	14	189,1	14	228,8	14
278,1	15	325,8	15	255,0	15	729,6	15	307,4	15	179,5	15
299,3	16	329,7	16	278,5	16	749,8	16	647,3	16	846,0	16
580,8	17	115,2	17	141,4	17	115,2	17	115,2	17	256,6	17
329,2	18	273,5	18	171,2	18	969,6	18	227,6	18	294,8	18
169,8	19	208,1	19	297,8	19	192,7	19	130,0	19	395,4	19
115,2	20	204,3	20	168,1	20	115,2	20	236,9	20	1134,0	20
1039,3	21	137,4	21	311,7	21	553,5	21	434,9	21	247,5	21
732,9	22	278,5	22	1398,5	22	446,0	22	520,2	22	463,4	22
390,9	23	168,2	23	930,8	23	575,9	23	233,5	23	913,5	23
265,0	24	826,1	24	261,7	24	688,8	24	583,8	24	455,3	24

S7 (Wh)	H	S8 (Wh)	H	S9 (Wh)	H	S10 (Wh)	H	PreuMM (c€/kWh)		
155,1	1	200,6	1	699,1	1	261,8	1	9,000	mc€/Wh	90
115,2	2	115,2	2	195,0	2	210,3	2	8,700	mc€/Wh	87
250,3	3	185,6	3	202,0	3	334,3	3	8,405	mc€/Wh	84,05
172,1	4	151,4	4	147,7	4	169,3	4	8,120	mc€/Wh	81,2
202,7	5	200,7	5	195,2	5	186,8	5	8,120	mc€/Wh	81,2
155,6	6	168,4	6	201,6	6	183,7	6	8,405	mc€/Wh	84,05
179,1	7	183,1	7	146,6	7	218,6	7	8,500	mc€/Wh	85
347,1	8	685,0	8	747,0	8	302,5	8	9,355	mc€/Wh	93,55
256,5	9	301,2	9	156,0	9	136,9	9	9,681	mc€/Wh	96,81
217,0	10	220,5	10	154,5	10	1204,6	10	9,200	mc€/Wh	92
455,7	11	296,3	11	115,2	11	364,3	11	9,000	mc€/Wh	90
801,4	12	269,9	12	218,1	12	192,3	12	9,200	mc€/Wh	92
115,2	13	730,4	13	154,2	13	171,5	13	9,200	mc€/Wh	92
266,9	14	177,7	14	184,8	14	166,3	14	9,200	mc€/Wh	92
236,0	15	131,7	15	879,9	15	452,9	15	9,110	mc€/Wh	91,1
115,2	16	273,5	16	115,2	16	192,2	16	8,951	mc€/Wh	89,51
115,2	17	256,8	17	1443,9	17	119,8	17	8,970	mc€/Wh	89,7
522,5	18	2540,0	18	115,2	18	839,6	18	9,220	mc€/Wh	92,2
143,5	19	444,3	19	413,2	19	2048,6	19	9,460	mc€/Wh	94,6
256,2	20	168,5	20	189,8	20	115,2	20	10,000	mc€/Wh	100
1401,2	21	218,9	21	985,9	21	208,4	21	10,000	mc€/Wh	100
115,2	22	580,1	22	2186,1	22	264,5	22	9,736	mc€/Wh	97,36
172,9	23	264,9	23	115,2	23	199,5	23	9,455	mc€/Wh	94,55
600,2	24	161,9	24	227,7	24	306,5	24	9,274	mc€/Wh	92,74

A5. Dades Escenari 2

S1 (Wh)	P1 (Wh)	S1- P1(Wh)	Costeh (mc€)	H	S2(Wh)	P2(Wh)	S2- P2(Wh)	Costeh (mc€)	H
652,7	0	652,7	5874,2	1	238,6	0	238,6	2147,3	1
436,9	0	436,9	3800,9	2	298,8	0	298,8	2599,5	2
173,4	0	173,4	1457,0	3	191,6	0	191,6	1610,5	3
227,9	0	227,9	1850,4	4	286,1	0	286,1	2322,8	4
154,4	0	154,4	1253,4	5	181,3	0	181,3	1471,9	5
191,0	0	191,0	1605,6	6	180,2	0	180,2	1514,2	6
277,0	0	277,0	2354,9	7	162,5	0	162,5	1381,0	7
1100,9	62	1038,9	9719,0	8	373,8	59	314,8	2945,2	8
461,1	206	255,1	2470,1	9	260,4	160	100,4	972,3	9
1044,7	285	759,7	6989,1	10	163,1	211	-47,9	-335,5	10
374,8	287	87,8	790,3	11	818,1	415	403,1	3627,7	11
560,2	465	95,2	876,1	12	224,5	627	-402,5	-2817,2	12
166,4	607	-440,6	-3084,2	13	203,0	735	-532,0	-3724,2	13
212,2	745	-532,8	-3729,3	14	218,1	894	-675,9	-4731,5	14
278,1	518	-239,9	-1679,2	15	325,8	696	-370,2	-2591,7	15
299,3	622	-322,7	-2258,6	16	329,7	639	-309,3	-2165,2	16
580,8	668	-87,2	-610,3	17	115,2	340	-224,8	-1573,6	17
329,2	570	-240,8	-1685,4	18	273,5	379	-105,5	-738,6	18
169,8	396	-226,2	-1583,6	19	208,1	228	-19,9	-139,0	19
115,2	218	-102,8	-719,6	20	204,3	76	128,3	1282,9	20
1039,3	54	985,3	9853,2	21	137,4	9	128,4	1284,0	21
732,9	0	732,9	7135,0	22	278,5	0	278,5	2711,1	22
390,9	0	390,9	3696,0	23	168,2	0	168,2	1590,4	23
265,0	0	265,0	2458,1	24	826,1	0	826,1	7661,2	24

S3(Wh)	P3(Wh)	S3-P3(Wh)	Costeh (mc€)	H	S4(Wh)	P4(Wh)	S4-P4(Wh)	Costeh (mc€)	H
266,3	0	266,3	2396,8	1	184,7	0	184,7	1662,3	1
267,6	0	267,6	2327,8	2	715,2	0	715,2	6222,0	2
174,6	0	174,6	1467,4	3	524,3	0	524,3	4406,7	3
178,9	0	178,9	1452,4	4	156,6	0	156,6	1271,9	4
169,0	0	169,0	1372,7	5	227,5	0	227,5	1846,9	5
175,9	0	175,9	1478,0	6	193,9	0	193,9	1629,4	6
140,2	0	140,2	1191,7	7	178,5	0	178,5	1517,4	7
741,1	11	730,1	6830,4	8	526,8	73	453,8	4245,5	8
377,3	77	300,3	2907,6	9	182,0	218	-36,0	-252,0	9
346,3	181	165,3	1520,3	10	125,5	431	-305,5	-2138,3	10
448,5	140	308,5	2776,7	11	230,9	599	-368,1	-2576,9	11
155,0	85	70,0	643,9	12	312,2	741	-428,8	-3001,7	12
115,2	205	-89,8	-628,6	13	168,9	843	-674,1	-4718,5	13
262,8	397	-134,2	-939,5	14	115,2	894	-778,8	-5451,6	14
255,0	700	-445,0	-3114,7	15	729,6	793	-63,4	-443,8	15
278,5	618	-339,5	-2376,8	16	749,8	730	19,8	177,5	16
141,4	630	-488,6	-3420,0	17	115,2	720	-604,8	-4233,6	17
171,2	373	-201,8	-1412,4	18	969,6	570	399,6	3684,6	18
297,8	295	2,8	26,7	19	192,7	396	-203,3	-1423,2	19
168,1	134	34,1	341,2	20	115,2	153	-37,8	-264,6	20
311,7	33	278,7	2787,1	21	553,5	38	515,5	5155,0	21
1398,5	0	1398,5	13615,3	22	446,0	0	446,0	4342,7	22
930,8	0	930,8	8801,2	23	575,9	0	575,9	5445,0	23
261,7	0	261,7	2426,6	24	688,8	0	688,8	6388,2	24

S5(Wh)	H	S6(Wh)	H	S7(Wh)	H	S8(Wh)	H
189,5	1	342,8	1	155,1	1	200,6	1
714,8	2	210,7	2	115,2	2	115,2	2
183,1	3	133,4	3	250,3	3	185,6	3
228,8	4	142,0	4	172,1	4	151,4	4
160,4	5	210,7	5	202,7	5	200,7	5
132,2	6	152,7	6	155,6	6	168,4	6
133,9	7	247,6	7	179,1	7	183,1	7
237,5	8	432,1	8	347,1	8	685,0	8
386,0	9	208,3	9	256,5	9	301,2	9
190,2	10	272,1	10	217,0	10	220,5	10
774,2	11	788,4	11	455,7	11	296,3	11
261,9	12	346,5	12	801,4	12	269,9	12
310,7	13	272,5	13	115,2	13	730,4	13
189,1	14	228,8	14	266,9	14	177,7	14
307,4	15	179,5	15	236,0	15	131,7	15
647,3	16	846,0	16	115,2	16	273,5	16
115,2	17	256,6	17	115,2	17	256,8	17
227,6	18	294,8	18	522,5	18	2540,0	18
130,0	19	395,4	19	143,5	19	444,3	19
236,9	20	1134,0	20	256,2	20	168,5	20
434,9	21	247,5	21	1401,2	21	218,9	21
520,2	22	463,4	22	115,2	22	580,1	22
233,5	23	913,5	23	172,9	23	264,9	23
583,8	24	455,3	24	600,2	24	161,9	24

S9(Wh)	H	S10(Wh)	H	PreuMM	
699,1	1	261,8	1	9,00000	mc€/Wh 90
195,0	2	210,3	2	8,70000	mc€/Wh 87
202,0	3	334,3	3	8,40500	mc€/Wh 84,05
147,7	4	169,3	4	8,12000	mc€/Wh 81,2
195,2	5	186,8	5	8,12000	mc€/Wh 81,2
201,6	6	183,7	6	8,40500	mc€/Wh 84,05
146,6	7	218,6	7	8,50000	mc€/Wh 85
747,0	8	302,5	8	9,35500	mc€/Wh 93,55
156,0	9	136,9	9	9,68100	mc€/Wh 96,81
154,5	10	1204,6	10	9,20000	mc€/Wh 92
115,2	11	364,3	11	9,00000	mc€/Wh 90
218,1	12	192,3	12	9,20000	mc€/Wh 92
154,2	13	171,5	13	9,20000	mc€/Wh 92
184,8	14	166,3	14	9,20000	mc€/Wh 92
879,9	15	452,9	15	9,11000	mc€/Wh 91,1
115,2	16	192,2	16	8,95100	mc€/Wh 89,51
1443,9	17	119,8	17	8,97000	mc€/Wh 89,7
115,2	18	839,6	18	9,22000	mc€/Wh 92,2
413,2	19	2048,6	19	9,46000	mc€/Wh 94,6
189,8	20	115,2	20	10,00000	mc€/Wh 100
985,9	21	208,4	21	10,00000	mc€/Wh 100
2186,1	22	264,5	22	9,73600	mc€/Wh 97,36
115,2	23	199,5	23	9,45500	mc€/Wh 94,55
227,7	24	306,5	24	9,27400	mc€/Wh 92,74

A6. Dades Escenari 3

S1	P1 (Wh)	S1-P1 (Wh)	S1-P1-EC(Wh)	Costeh(mc€)	H	S2 (Wh)	P2 (Wh)	S2-P2 (Wh)	S2-P2-EC(Wh)	Costeh (mc€)	H
652,7	0	652,7	652,7	5874,2	1	238,6	0	238,6	238,6	2147,3	1
436,9	0	436,9	436,9	3800,9	2	298,8	0	298,8	298,8	2599,5	2
173,4	0	173,4	173,4	1457,0	3	191,6	0	191,6	191,6	1610,5	3
227,9	0	227,9	227,9	1850,4	4	286,1	0	286,1	286,1	2322,8	4
154,4	0	154,4	154,4	1253,4	5	181,3	0	181,3	181,3	1471,9	5
191,0	0	191,0	191,0	1605,6	6	180,2	0	180,2	180,2	1514,2	6
277,0	0	277,0	277,0	2354,9	7	162,5	0	162,5	162,5	1381,0	7
1100,9	62	1038,9	1038,9	9719,0	8	373,8	59	314,8	314,8	2945,2	8
461,1	206	255,1	250,8	2464,2	9	260,4	160	100,4	98,7	970,0	9
1044,7	285	759,7	675,4	6896,4	10	163,1	211	-47,9	0,0	-388,2	10
374,8	287	87,8	78,8	781,3	11	818,1	415	403,1	361,8	3586,4	11
560,2	465	95,2	60,1	837,4	12	224,5	627	-402,5	0,0	-3259,9	12
166,4	607	-440,6	0,0	-3568,9	13	203,0	735	-532,0	0,0	-4309,4	13
212,2	745	-532,8	0,0	-4064,5	14	218,1	894	-675,9	0,0	-5156,8	14
278,1	518	-239,9	0,0	-1932,3	15	325,8	696	-370,2	0,0	-2982,3	15
299,3	622	-322,7	0,0	-2573,3	16	329,7	639	-309,3	0,0	-2466,9	16
580,8	668	-87,2	0,0	-696,2	17	115,2	340	-224,8	0,0	-1795,0	17
329,2	570	-240,8	0,0	-1952,7	18	273,5	379	-105,5	0,0	-855,8	18
169,8	396	-226,2	0,0	-1861,8	19	208,1	228	-19,9	0,0	-163,4	19
115,2	218	-102,8	0,0	-873,8	20	204,3	76	128,3	120,3	1270,9	20
1039,3	54	985,3	985,3	9853,2	21	137,4	9	128,4	128,4	1284,0	21
732,9	0	732,9	732,9	7135,0	22	278,5	0	278,5	278,5	2711,1	22
390,9	0	390,9	390,9	3696,0	23	168,2	0	168,2	168,2	1590,4	23
265,0	0	265,0	265,0	2458,1	24	826,1	0	826,1	826,1	7661,2	24

S3 (Wh)	P3 (Wh)	S3-P3 (Wh)	S3-P3-EC(Wh)	Costeh (mc€)	H	S4 (Wh)	P4 (Wh)	S4-P4 (Wh)	S4-P4-EC(Wh)	Costeh (mc€)	H
266,3	0	266,3	266,3	2396,8	1	184,7	0	184,7	184,7	1662,3	1
267,6	0	267,6	267,6	2327,8	2	715,2	0	715,2	715,2	6222,0	2
174,6	0	174,6	174,6	1467,4	3	524,3	0	524,3	524,3	4406,7	3
178,9	0	178,9	178,9	1452,4	4	156,6	0	156,6	156,6	1271,9	4
169,0	0	169,0	169,0	1372,7	5	227,5	0	227,5	227,5	1846,9	5
175,9	0	175,9	175,9	1478,0	6	193,9	0	193,9	193,9	1629,4	6
140,2	0	140,2	140,2	1191,7	7	178,5	0	178,5	178,5	1517,4	7
741,1	11	730,1	730,1	6830,4	8	526,8	73	453,8	453,8	4245,5	8
377,3	77	300,3	295,2	2900,7	9	182,0	218	-36,0	0,0	-300,3	9
346,3	181	165,3	146,9	1500,1	10	125,5	431	-305,5	0,0	-2474,4	10
448,5	140	308,5	276,9	2745,0	11	230,9	599	-368,1	0,0	-2945,1	11
155,0	85	70,0	44,2	615,5	12	312,2	741	-428,8	0,0	-3473,4	12
115,2	205	-89,8	0,0	-727,4	13	168,9	843	-674,1	0,0	-5460,0	13
262,8	397	-134,2	0,0	-1024,0	14	115,2	894	-778,8	0,0	-5941,6	14
255,0	700	-445,0	0,0	-3584,2	15	729,6	793	-63,4	0,0	-510,7	15
278,5	618	-339,5	0,0	-2708,0	16	749,8	730	19,8	11,1	168,9	16
141,4	630	-488,6	0,0	-3901,3	17	115,2	720	-604,8	0,0	-4829,3	17
171,2	373	-201,8	0,0	-1636,4	18	969,6	570	399,6	355,3	3635,3	18
297,8	295	2,8	2,5	26,3	19	192,7	396	-203,3	0,0	-1673,3	19
168,1	134	34,1	32,0	338,0	20	115,2	153	-37,8	0,0	-321,3	20
311,7	33	278,7	278,7	2787,1	21	553,5	38	515,5	515,5	5155,0	21
1398,5	0	1398,5	1398,5	13615,3	22	446,0	0	446,0	446,0	4342,7	22
930,8	0	930,8	930,8	8801,2	23	575,9	0	575,9	575,9	5445,0	23
261,7	0	261,7	261,7	2426,6	24	688,8	0	688,8	688,8	6388,2	24

S5(Wh)	S5-EC(Wh)	Costeh (mc€)	H	S6(Wh)	S6-EC(Wh)	Costeh	H	S7(Wh)	S7-EC(Wh)	Costeh (mc€)	H
189,5	189,5	1705,8	1	342,8	342,8	3085,0	1	155,1	155,1	1395,5	1
714,8	714,8	6218,3	2	210,7	210,7	1832,9	2	115,2	115,2	1002,2	2
183,1	183,1	1539,3	3	133,4	133,4	1120,9	3	250,3	250,3	2103,9	3
228,8	228,8	1857,9	4	142,0	142,0	1152,7	4	172,1	172,1	1397,2	4
160,4	160,4	1302,1	5	210,7	210,7	1711,3	5	202,7	202,7	1645,6	5
132,2	132,2	1111,3	6	152,7	152,7	1283,7	6	155,6	155,6	1307,6	6
133,9	133,9	1138,3	7	247,6	247,6	2104,7	7	179,1	179,1	1522,0	7
237,5	237,5	2221,6	8	432,1	432,1	4042,5	8	347,1	347,1	3247,4	8
386,0	379,4	3728,3	9	208,3	204,7	2011,3	9	256,5	252,1	2477,1	9
190,2	169,1	1726,8	10	272,1	241,9	2470,3	10	217,0	192,9	1969,6	10
774,2	694,9	6888,3	11	788,4	707,6	7014,9	11	455,7	409,0	4054,7	11
261,9	165,3	2303,1	12	346,5	218,8	3047,1	12	801,4	506,0	7048,0	12
310,7	3,2	2520,4	13	272,5	2,8	2210,0	13	115,2	1,2	934,4	13
189,1	-141,5	1531,4	14	228,8	-171,2	1853,6	14	266,9	-199,7	2162,1	14
307,4	150,2	2634,5	15	179,5	87,7	1538,1	15	236,0	115,3	2022,7	15
647,3	362,7	5516,5	16	846,0	474,0	7209,4	16	115,2	64,5	981,7	16
115,2	45,0	964,2	17	256,6	100,3	2147,6	17	115,2	45,0	964,2	17
227,6	202,3	2070,0	18	294,8	262,1	2682,0	18	522,5	464,5	4753,3	18
130,0	113,7	1210,1	19	395,4	345,8	3679,7	19	143,5	125,5	1335,5	19
236,9	222,2	2347,2	20	1134,0	1063,5	11233,9	20	256,2	240,2	2537,7	20
434,9	434,9	4349,5	21	247,5	247,5	2475,4	21	1401,2	1401,2	14011,8	21
520,2	520,2	5064,3	22	463,4	463,4	4511,2	22	115,2	115,2	1121,6	22
233,5	233,5	2208,2	23	913,5	913,5	8636,9	23	172,9	172,9	1634,5	23
583,8	583,8	5414,5	24	455,3	455,3	4222,1	24	600,2	600,2	5566,1	24



S8(Wh)	S8-EC(Wh)	Costeh (mc€)	H	S9(Wh)	S9-EC(Wh)	Costeh (mc€)	H	S10(Wh)	S10-EC(Wh)	Costeh (mc€)
200,6	200,6	1805,4	1	699,1	699,1	6291,7	1	261,8	261,8	2356,3
115,2	115,2	1002,2	2	195,0	195,0	1696,3	2	210,3	210,3	1829,6
185,6	185,6	1559,6	3	202,0	202,0	1697,9	3	334,3	334,3	2810,2
151,4	151,4	1229,1	4	147,7	147,7	1199,3	4	169,3	169,3	1374,9
200,7	200,7	1629,6	5	195,2	195,2	1585,2	5	186,8	186,8	1516,6
168,4	168,4	1415,0	6	201,6	201,6	1694,5	6	183,7	183,7	1544,0
183,1	183,1	1556,1	7	146,6	146,6	1246,5	7	218,6	218,6	1857,9
685,0	685,0	6408,6	8	747,0	747,0	6988,3	8	302,5	302,5	2830,1
301,2	296,0	2908,6	9	156,0	153,3	1506,2	9	136,9	134,6	1322,5
220,5	196,0	2001,4	10	154,5	137,3	1402,5	10	1204,6	1070,9	10935,2
296,3	266,0	2636,7	11	115,2	103,4	1025,0	11	364,3	327,0	3241,3
269,9	170,4	2373,5	12	218,1	137,7	1918,0	12	192,3	121,4	1691,0
730,4	7,5	5924,5	13	154,2	1,6	1250,9	13	171,5	1,8	1391,1
177,7	-132,9	1439,0	14	184,8	-138,3	1496,7	14	166,3	-124,5	1347,1
131,7	64,3	1128,5	15	879,9	430,0	7541,3	15	452,9	221,3	3881,3
273,5	153,3	2331,1	16	115,2	64,5	981,7	16	192,2	107,7	1638,2
256,8	100,4	2149,2	17	1443,9	564,5	12085,6	17	119,8	46,9	1003,1
2540,0	2258,2	23106,3	18	115,2	102,4	1048,0	18	839,6	746,4	7637,8
444,3	388,5	4134,5	19	413,2	361,3	3845,0	19	2048,6	1791,3	19063,6
168,5	158,0	1669,2	20	189,8	178,0	1880,3	20	115,2	108,0	1141,3
218,9	218,9	2189,5	21	985,9	985,9	9858,6	21	208,4	208,4	2084,4
580,1	580,1	5648,2	22	2186,1	2186,1	21283,8	22	264,5	264,5	2575,3
264,9	264,9	2504,4	23	115,2	115,2	1089,2	23	199,5	199,5	1886,7
161,9	161,9	1501,6	24	227,7	227,7	2111,5	24	306,5	306,5	2842,6

H	PreuMM		PreuEauth		ConsumsT	ConsumsP	ProduccióT	Excedentso	ExcedentsF	Preu Ex	
1	9,00000	mc€/Wh	8,00000	mc€/Wh	3191	3191	0	0	0	7,00	90
2	8,70000	mc€/Wh	7,85000	mc€/Wh	3280	3280	0	0	0	7,00	87
3	8,40500	mc€/Wh	7,70250	mc€/Wh	2353	2353	0	0	0	7,00	84,05
4	8,12000	mc€/Wh	7,56000	mc€/Wh	1861	1861	0	0	0	7,00	81,2
5	8,12000	mc€/Wh	7,56000	mc€/Wh	1889	1889	0	0	0	7,00	81,2
6	8,40500	mc€/Wh	7,70250	mc€/Wh	1735	1735	0	0	0	7,00	84,05
7	8,50000	mc€/Wh	7,75000	mc€/Wh	1867	1867	0	0	0	7,00	85
8	9,35500	mc€/Wh	8,17750	mc€/Wh	5494	5289	205	0	0	7,00	93,55
9	9,68100	mc€/Wh	8,34050	mc€/Wh	2726	2101	661	-36	0	7,00	96,81
10	9,20000	mc€/Wh	8,10000	mc€/Wh	3938	3184	1108	-353	0	7,00	92
11	9,00000	mc€/Wh	8,00000	mc€/Wh	4666	3594	1441	-368	0	7,00	90
12	9,20000	mc€/Wh	8,10000	mc€/Wh	3342	2255	1918	-831	0	7,00	92
13	9,20000	mc€/Wh	8,10000	mc€/Wh	2408	1755	2390	-1737	0	7,00	92
14	9,20000	mc€/Wh	8,10000	mc€/Wh	2022	1214	2930	-2122	-908	7,00	92
15	9,11000	mc€/Wh	8,05500	mc€/Wh	3776	2187	2707	-1118	0	7,00	91,1
16	8,95100	mc€/Wh	7,97550	mc€/Wh	3847	2209	2609	-972	0	7,00	89,51
17	8,97000	mc€/Wh	7,98500	mc€/Wh	3260	2307	2358	-1405	0	7,00	89,7
18	9,22000	mc€/Wh	8,11000	mc€/Wh	6283	4939	1892	-548	0	7,00	92,2
19	9,46000	mc€/Wh	8,23000	mc€/Wh	4444	3578	1315	-449	0	7,00	94,6
20	10,00000	mc€/Wh	8,50000	mc€/Wh	2703	2263	581	-141	0	7,00	100
21	10,00000	mc€/Wh	8,50000	mc€/Wh	5539	5405	134	0	0	7,00	100
22	9,73600	mc€/Wh	8,36800	mc€/Wh	6985	6985	0	0	0	7,00	97,36
23	9,45500	mc€/Wh	8,22750	mc€/Wh	3965	3965	0	0	0	7,00	94,55
24	9,27400	mc€/Wh	8,13700	mc€/Wh	4377	4377	0	0	0	7,00	92,74

