TÉCNICO LISBOA

Blockchain-based Smart Contracts Application for Energy Trading

Joana Margarida Morais Bugalho

Thesis to obtain the Master of Science Degree in

Engineering Physics

Supervisors: Prof. Carlos Augusto Santos Silva
Prof. Ilídio Pereira Lopes

Examination Committee

Chairperson: Prof. Maria Teresa Haderer de la Peña Stadler
Supervisor: Prof. Carlos Augusto Santos Silva
Member of the Committee: Prof. Diana Pereira Neves

October 2022

"Change might not be fast and it isn't always easy.
But with time and effort, almost any habit can be reshaped."
in *The Power of Habit* by Charles Duhigg

Acknowledgments

This thesis would not be complete without the people that accompanied me throughout this journey.

First and foremost, I would like to thank my supervisors for the trust they have placed in me, especially to Professor Carlos Silva. With your guidance, I was able to work in a field that inspired me.

To my university friends, a special thanks for always supporting me. I am deeply indebted, particularly to Mariana Caseiro and Sofia Costa, for always being there for me. Both of you got me through this five-year journey at IST with a smile on my face.

I am also grateful to all of my close friends, that were with me every step of the way, even through late-night conversations when I asked for advice. Your belief in me kept me motivated all the way.

Last but not least, words cannot express my gratitude to my close family. You show me every day how strong I can be.

I could not have undertaken this project without your support. And for that, all of you have my deepest thanks.

Resumo

O mundo está longe de atingir os objetivos da Agenda 2030 para o Desenvolvimento Sustentável. O setor de energia é responsável por quase três quartos das emissões que causam as alterações climáticas, provando que, para atingir estes objetivos, é necessário implementar novas soluções neste setor.

É imperativo reestruturar o setor energético para acomodar uma maior parcela de fontes de energia sustentáveis, especialmente no setor residencial, uma vez que é um dos maiores consumidores de eletricidade no mundo.

Com a transição para as energias renováveis, um novo tipo de agentes que consomem energia, mas também conseguem produzir - os *prosumers* - passam a ter um papel mais importante no mercado. Por isso, a estrutura tradicional centralizada do mercado não consegue suportar estes novos desafios. É crucial criar novos modelos de comercialização de energia, que sejam descentralizados, eficientes e seguros.

Esta tese propõe o desenvolvimento de uma aplicação, utilizando tecnologia *blockchain*, que implementa contratos inteligentes que validam e auditam automaticamente as transações de energia.

Três contratos inteligentes foram implementados, usando a *blockchain Ethereum*: transações exclusivamente com a Empresa de Serviços de Energia, transações entre pares de acordo com regras predefinidas e transações entre pares através de leilões.

Embora os contratos inteligentes usando tecnologia blockchain serem uma opção viável para transações de energia, a blockchain Ethereum não é a melhor *blockchain* para negociar pequenas quantidades de energia, já que as taxas aplicadas durante o dia são muito mais altas comparadas com o preço pago pela energia.

Palavras-chave: Contractos Inteligentes, Tecnologia Blockchain, Troca de Energia Descentralizada, Transação, Comunidade Inteligente de Energia

Abstract

The world is far from achieving the goals set out by the 2030 Agenda for Sustainable Development. The energy sector is now responsible for almost three-quarters of the emissions causing climate change, proving that, to achieve these goals, the implementation of new solutions in this sector is essential.

It is imperative to restructure the energy sector to accommodate a bigger share of sustainable energy sources, more specifically in the residential sector, since it has one of the biggest electricity consumption in the world.

With the shift toward renewable energy, new agents that consume but can also produce - the prosumers - start to have a more important role in the market. For this reason, the traditional centralized structure of the market can no longer support the new challenges. It becomes more crucial to create new models for energy trading, that are decentralized, efficient and running on secure platforms.

This thesis proposes the development of an application, using blockchain technology, that implements smart contracts that automatically validate and audit energy transactions.

Three different smart contracts were implemented, using the Ethereum blockchain: trading exclusively with the Energy Service Company (ESCO), peer-to-peer (P2P) trading according to predefined rules and P2P trading through auctions.

While blockchain-based smart contracts proved to be a viable option for energy trading, the Ethereum blockchain is not the best blockchain to trade small amounts of energy, since the fees applied during the day were much higher compared to the price paid for the energy.

Keywords: Smart Contracts, Blockchain Technology, Decentralized Energy Trading, Transactions, Smart Energy Communities

Contents

List of Tables

List of Figures

List of Abbreviations

API	Application Programming Interface
dApp	Decentralized Application
ESCO	Energy Service Company
ETH	Ether (Ethereum cryptocurrency)
EVM	Ethereum Virtual Machine
EU	European Union
EOA	Externally Owned Accounts
GWEI	Gigawei
HMS	Home Management System
NEPC	National Energy and Climate Plans
P2P	Peer-to-Peer
PV	Photovoltaic
PBFT	Practical Byzantine Fault Tolerance
PoA	Proof of Authority
PoS	Proof of Stake
PoW	Proof of Work
SC	Smart Contract
UI	User Interface

Chapter 1

Introduction

The need to restructure the energy market has become a pressing matter in recent years. As the energy market evolves to integrate more sustainable energy sources, new agents that consume but can also produce energy - the so called prosumers - start to participate in the market. Consequently, the traditional centralized structure solution can no longer support the challenges that the energy market brings today. It becomes crucial to create new business models, decentralized, efficient and running on secure platforms, to support energy interactions within a community, such that the use of sustainable energy becomes more affordable and reliable.

Smart contracts, programs used to automate the execution of an agreement, without the need for a third-party, along with blockchain technology, that stores all the information in a decentralized manner, can be ideal for the interactions that occur within an energy community. All participants would be able to trade energy between them, according to their own preferences.

This dissertation proposes the development of a platform, using a blockchain-based solution, that implements smart contracts that automatically validate and audit energy transitions, contributing in this way to a more sustainable society. With this system, several activities can be automated: defining electricity costs for specific periods, different payment policies, defining schedules for buying and selling electricity, settlements details, etc..

Different smart contracts will be designed and implemented, then tested in an energy community of four prosumers and one Energy Service Company (ESCO). The smart contracts will be studied, along with blockchain technology, to determine which is the best option for energy trading inside a local energy community.

Apart from contributing to the reduction of carbon emissions and increase in energy efficiency, it may also have significant financial impacts, by shielding the market from fossil fuels market instability. Therefore, compared to the systems in place today, we can increase the efficiency, speed and scalability of the energy markets.

In the following subsections, it is given the context as to why new platforms for energy trading are necessary and a solution is proposed.

1.1 Motivation

Despite significant progress over the last two decades, the world is still falling short in the 2030 Agenda for Sustainable Development and the 17 Sustainable Development Goals (SDGs) [1], adopted by all United Nations Member States in 2015, which address economic, social, and environmental challenges. There are in particular 3 goals that are relevant for the energy sector: ensuring access to sustainable energy; developing sustainable consumption and production patterns; and taking action against climate change and carbon emissions.

Renewable energy used in the energy sector has increased. The share of consumption of renewable energy was estimated to be around 22% in 2021, with the main contribution from the electricity sector, with help from the heat and transport sectors [2]. However, 10% of the world population still lacks access to electricity.

Energy efficiency, the shift towards renewable energy and the development and deployment of low-carbon process routes are all critical to reach a sustainable system and to mitigate the climate crisis. To meet the 2030 Agenda, the improvement rate needs to quicken, as the increase in energy demand will now need to average 3% a year up to 2030, compared to the previous 2% [3]. This is still possible, yet a significant investment is essential. Governments can accelerate the progress by reducing risks associated with developing new technologies and adopting mandatory CO_2 emissions reduction and energy efficiency policies [4].

Global CO_2 emissions declined by 5.8% in 2020, almost 2 Gt CO_2, mostly due to the pandemic impact in the world economy, which decreased the demand for oil and coal harder than other energy sources, while renewables increased. Despite the decline in 2020, global energy-related CO_2 emissions remained at 31.5 Gt CO_2, which contributed to CO_2 reaching its highest-ever average annual concentration in the atmosphere of 412.5 parts per million in 2020 (around 50% higher than when the industrial revolution began). After the pandemic, the global CO_2 emissions rose once again in 2021, reaching 33.0 Gt CO_2 [5], demonstrating that these emissions will continue to rise, if nothing is done to reverse it.

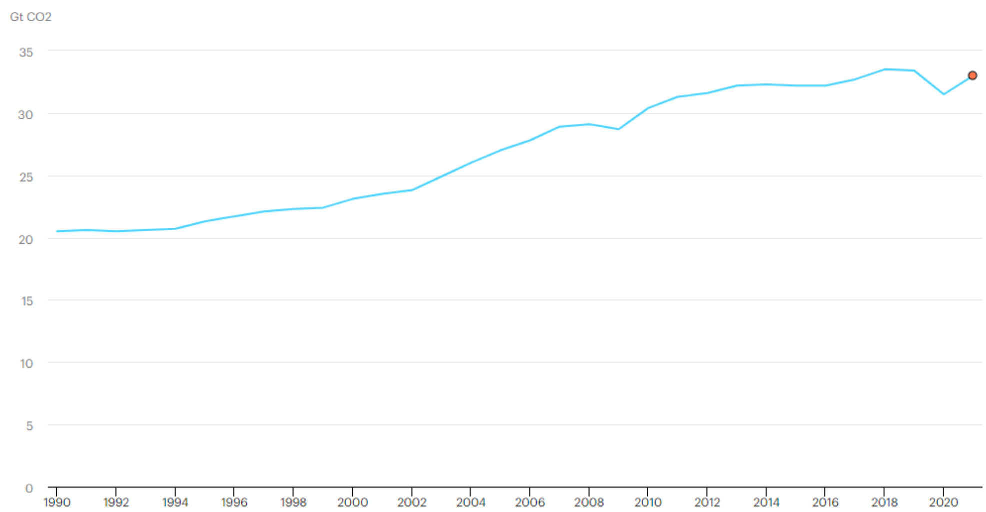

Figure 1.1: Global energy-related CO_2 emissions from 1990 to 2021 [5].

Over the last years, there has been a growth in new installations of renewable electricity, especially in developing countries, and studies show that renewable energy capacity will continue to expand, contributing to the reduction of carbon emissions. However, renewable energy is still underused in most countries, and as most of the energy generation is still based on fossil fuels, greenhouse gas emissions will continue to increase, inducing global warming and consequently climate change. The effort to eliminate fossil-fuel subsidies is uneven, discouraging once again the development of clean and renewable energy [3].

Amidst the increase in energy consumption, the energy sector is now responsible for almost three-quarters of the emissions causing climate change, proving that the energy sector has to be at the heart of the solution to climate change [6].

To decrease greenhouse gas emissions, mostly carbon dioxide due to the burning of fossil fuels, and to mitigate the effects of climate change, a bigger shift in the economy towards carbon neutrality is required [2], based on increasing levels of energy efficiency and renewable energy resources.

This confirms that the future of energy also lies with homeowners and residential communities. Renewable energy generation must become more inviting in the residential sector, so that carbon neutrality becomes a reality within reach. Now that the installation of renewable energy is growing, it is fundamental to create the right tools to make this energy market more attractive and affordable.

1.2 Conceptual Model

The main goal of this thesis is to create a successful framework for energy trading in communities, more specifically a blockchain-based smart contracts application (Figure 1.2). This solution should provide an efficient and secure way to exchange energy between participants, who want to engage in the energy market, focusing primarily on building a platform where the participants can trade tokens for the energy.

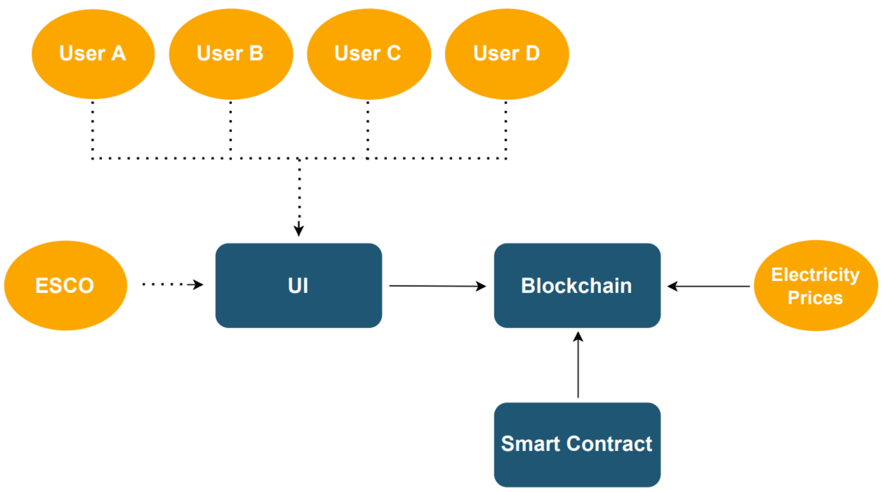

Figure 1.2: Proposed conceptual diagram for the proposed solution.

All participants (users) will be connected to the energy trading application through an user interface (UI), where they can interact with each other. An Energy Service Company (ESCO) will also be a part of this community, to guarantee that there is always enough energy inside the community. This ESCO will buy or sell energy to the community when needed, and it will provide all energy-related equipment to assure energy trading operations. Additionally, each user can decide if they wish to use smart meters and a home management system. The platform is then connected to a blockchain, which has a smart contract stored within. Since there is an ESCO in this community, that will acquire energy when needed, the blockchain is also connected to the present electricity costs.

In Figure 1.3, there is the representation of the architecture for the proposed solution. Users will inform the smart contract if they wish to buy or sell energy, then the smart contract will compute the price at which the electricity will be sold at that moment. The price will be exchanged between buyer and seller, in the form of tokens (cryptocurrency). After the transaction is complete, the electricity is sent through the infrastructure shared between the community.

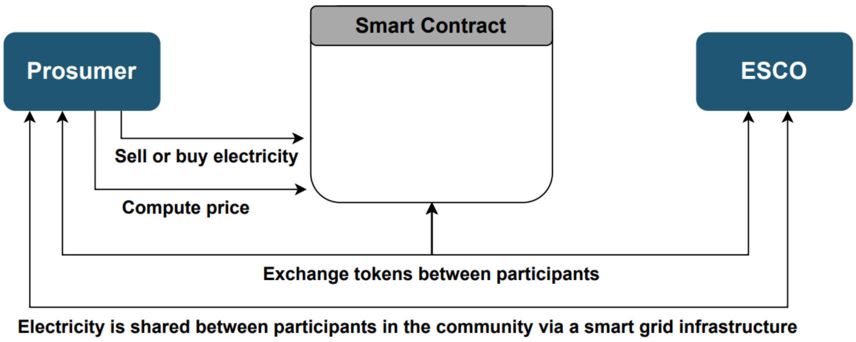

Figure 1.3: Architecture of the proposed solution.

The design of the smart grid infrastructure is not within the scope of this project, but note that this structure needs to connect all participants in this energy trading community, and it also needs to have devices connected to the application in each participant's house, to account for the flow of electricity.

1.3 Methodical Approach

To create such an energy trading application, first it is necessary to lay out the steps that play into it. The milestones for the conclusion of this project were the following:

1. Greater understanding of the matters that comprise this thesis, such as energy trading communities and energy markets, smart contracts and blockchain technology.
2. Define the type of participants in energy communities.
3. Define the testing phase of the application, to make sure that the most relevant situations are accounted for, when designing the application.
4. Develop a generic smart contract, in order to identify and characterize the main components in the energy contract, like respective costs and/or revenues.

5. Choose a public blockchain framework, one that can possibly store the smart contract data from all participants and implement contract verification and validation.
6. Understand the complexity of smart contracts for energy trading, and define steps to achieve the best smart contract possible.
7. For each of the steps, create a smart contract and integrate it within the blockchain.
8. Design a user interface, to connect participants with the blockchain.
9. Test the application for each of the smart contracts deployed, according to the agreed test runs.
10. Discuss the results obtained, and the main components of the application.
11. Discuss the challenges that need to be addressed to implement this application in the real world.
12. Consider the limitations of this application and possible future developments.

With the application developed, the system should be user-friendly and highly scalable, i.e., should be implemented in any type of energy communities, helping in this way, the energy market to transition towards a more sustainable system.

1.4 Thesis Outline

This dissertation is divided into 6 different chapters:

1. **Introduction**: this dissertation begins with the motivation, the conceptual model of the solution, the methodical approach of this project and the outline of this paper.
2. **Background and Literature Review**: this chapter focuses first on laying out the current situation of the energy sector, followed by all the relevant topics necessary for the proposed solution, such as smart contracts and blockchains, and the current development in this field.
3. **Methodology**: this chapter describes the energy community, the type of participants and how they can interact with each other. Then, the proposed solutions for the smart contracts are explained.
4. **Implementation and Results**: this chapter reveals how the application was implemented, giving a brief insight into how to achieve the solution. After, the results acquired when testing are presented.
5. **Discussion**: in this chapter, the results are discussed along with the main components of the application.
6. **Concluding Remarks**: this chapter presents a succinct description of the work done, the achievements, the limitations and possible improvements of the proposed solution.

Chapter 2

Background and Literature Review

2.1 Energy Sector

The world's ambition is to achieve Net Zero Carbon Emissions by 2050. Since the energy sector represents almost 75% of those emissions, this sector needs to change drastically, focusing primarily on increasing renewable energy generation and consumption, and expand energy efficiency and digitalization. For that, it becomes also crucial to develop new decentralized structures for the energy market, that support the interactions the prosumers bring.

2.1.1 Energy Production and Consumption

Energy markets are evolving, with efforts to make them more sustainable. Renewable sources of energy, such as solar and wind, are starting to have a more significant role in these systems. Even so, this revolution still has a long way to go [6].

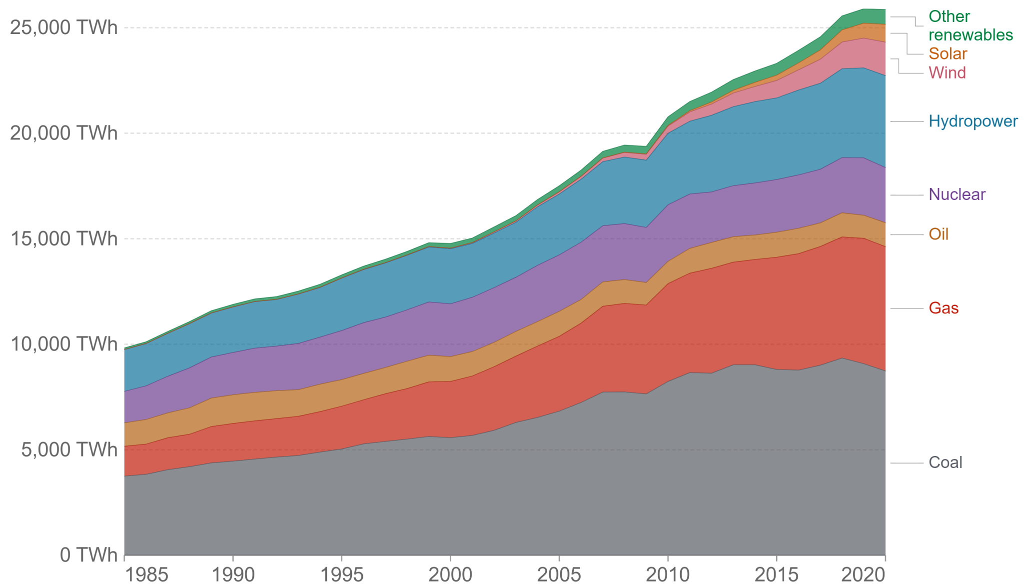

Figure 2.1: World final electricity production by source [7].

As it can be seen in Figure 2.1, the production of renewable energy resources has been increasing over the last few years. In 2020, renewable electricity generation rose about 7%, with the share of renewables in global electricity generation reaching almost 29%, an annual record, causing an increase of 3% in the renewable energy use [8]. Even so, renewable energy deployment as a whole needs to expand significantly to meet the Net Zero Emissions, a share of more than 60% of renewable energy generation is necessary by 2030 [8].

On the other hand, the world's total electricity consumption continues to rise significantly, with an increase of 4.5% from 2020 to 2021 [9]. In Figure 2.2, the share of electricity final consumption is represented by sector. The residential consumption of electricity had a high share of 26.6% in 2019, only topped by the industrial sector with 41.9% [10], revealing that the residential sector has to be adjusted to include more renewable energy.

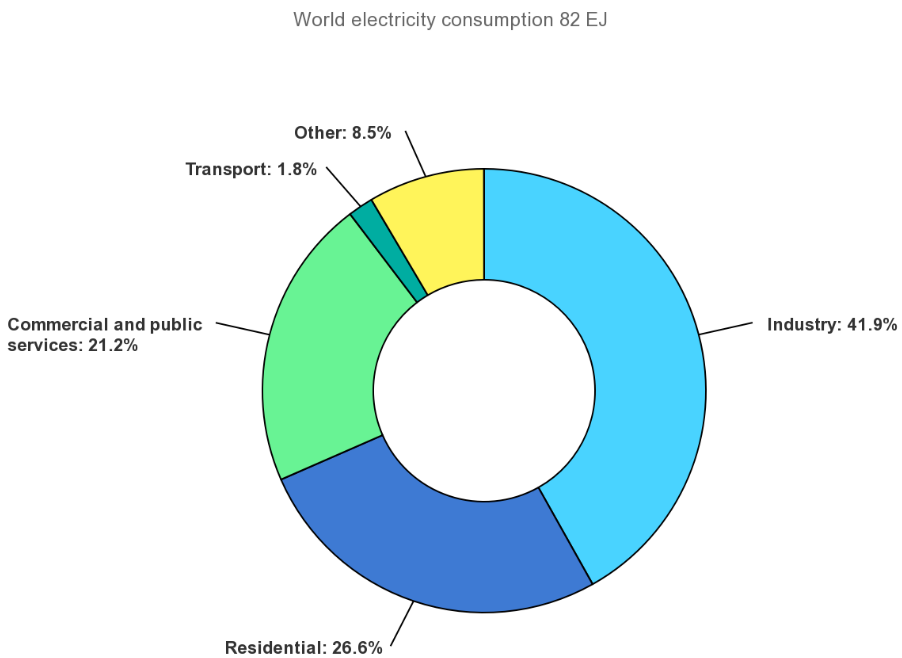

Figure 2.2: Share of electricity final consumption by sector, in 2019 [10].

If nothing changes, renewable generation will continue to grow, but it cannot keep up with the increasing demand. Thus, it is also fundamental to address other efforts for energy consumption reduction, such as energy efficiency.

2.1.2 Energy Efficiency

The energy efficiency progress is ongoing, but is lacking financial aid. To meet the Net Zero by 2050, the investment in this area needs to double [11]. Energy efficiency offers some of the fastest and most cost-effective actions to reduce CO_2 emissions. If energy consumption is reduced, the share of renewable energy sources will inevitably increase and outpace overall demand for energy services. In this scenario, the global economy will grow 40% by 2030, driven by higher populations and income levels, but it will use 7% less energy [11].

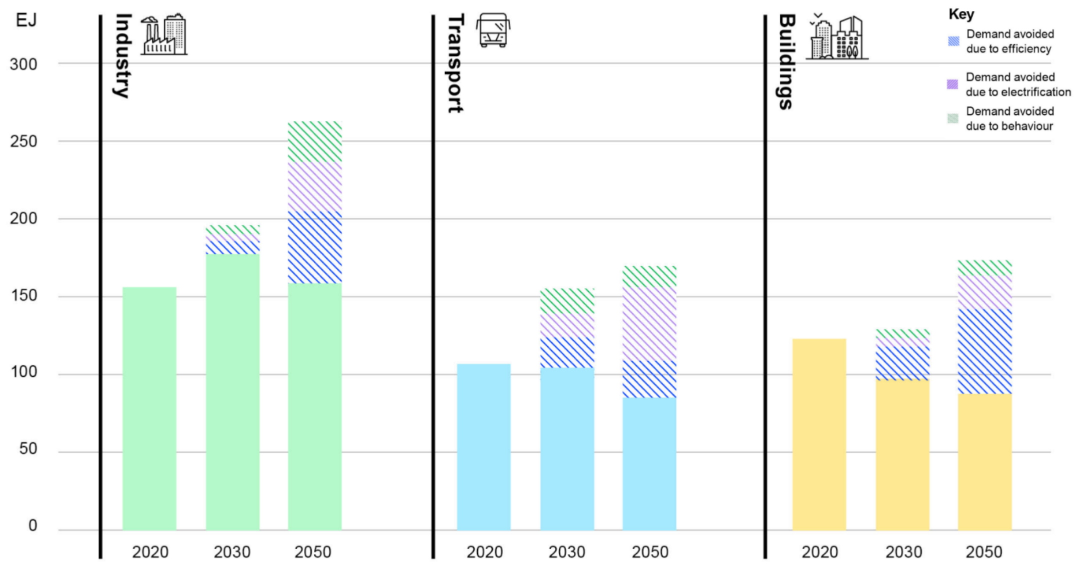

Figure 2.3: Energy efficiency milestones in the Net Zero Emissions by 2050 Scenario [11].

In Figure 2.3, it is shown how energy efficiency can change energy consumption globally, in all main energy sectors. And even though energy consumption is expected to continue to rise in the next three decades, it is possible to reduce energy consumption by applying simple energy efficiency strategies (represented in blue in the graphic).

While reducing energy consumption and consequently carbon emissions, energy efficiency can also have a compelling financial impact.

Government policies are expected to help energy efficiency investment to rise by 10% in 2021 to almost USD 300 billion [11], but once again the world is falling short to reach Net Zero Emissions. In recent years, the investments have grown significantly in Europe, still energy efficiency requires a much larger investment.

In July 2021, the International Energy Agency, IEA, released the tool *Sustainable Recovery Tracker*, to assess the return of the investments made in the energy sector. That tracker indicated that the annual average investments, both public and private investments, are likely to increase clean energy and sustainable recovery spending around 400 USD billion annually between 2021 and 2023. Of this annual investments, almost USD 260 billion per year is expected for energy efficiency measures in buildings, industry and low-carbon transport [11].

The total governmental approved energy efficiency investments in 2021 to 2023 comprise over USD 310 to 315 billion. The European Union countries account for around 65% of this total, with 55% going to the transport sector, 30% to the residential sector, 10% to the industry sector and the remaining 15% to other measures [11].

As more countries begin to understand the power that energy efficiency has on energy consumption, energy efficiency policies also begin to grow stronger in terms of implementation. Energy efficiency starts to have a critical role in the policies made to reduce greenhouse gas emissions. Mandatory minimum energy efficiency performance standards and/or energy labels are now required in several equipments in the commercial, industrial and residential sectors, such as air conditioner, refrigeration, washing machines, lighting, industrial motors and passenger cars.

2.1.3 Digitalization in the energy sector

Another way to increase substantially energy efficiency is digitalization. Digitalization allows the gathering of data that consequently can bring interventions that contribute to energy efficiency.

Prosumers - consumers that are also producers - are starting to emerge in energy communities. These new agents have access to a renewable source of energy that can be exploited for personal use and/or can also be sold to another person directly – peer-to-peer trading – or to an aggregator or retailer operating in energy markets. These communities promote the active participation of citizens in the energy system [12].

This evolution requires that electricity grids become smart grids, that are managed and operated in real-time, taking advantage of the use of information and communication technologies, like smart meters that collect energy data in real-time.

Through digitalization, energy efficiency has a massive potential to evolve, based on technologies that collect and analyze data. Then, the data is used to make changes in the real world, through automatization or human intervention. Digital technologies in the energy sector can boost energy efficiency with three steps: data gathering, data analysis and action.

Data gathering contributed immensely to the digital era, with data proliferation [13]. Nowadays, everywhere around the world, 2.5 quintillion bytes of data are being generated every day. For example, global Internet traffic generated 1.5 zettabytes of data in 2018, 19 times more than a decade ago [14]. And it is only expected to increase even more with digitalization.

From this wide range of data, there is a considerable portion that is directly relevant to the energy sector, more specifically to energy efficiency. Not only data on energy consumption and production is important, but also the conditions for that period of time, like weather conditions. This data can be extracted through smart meters, data from mobile telephones or online networks.

Usually, in the energy sector, relevant data is generated and gathered through sensors or meters. Sensors measure conditions from the physical environment, such as temperature and light. Meters capture and send information on energy consumption or production to the user and utilities or grid operators automatically. More specifically, smart meters can capture high-resolution information in real-time, such as energy usage, reverse flows and other factors. These more advanced meters are becoming universally deployed, not only for granting detailed analyses of the energy demand and efficiency opportunities, but also for facilitating communication between the consumer and the energy retailer.

While sensors and meters are not recent technology, they are not commonly installed in the households of consumers yet. Nevertheless, they are beginning to spread rapidly (Figure 2.4), thanks to lower costs, better performance, size reductions and government support programs. By 2017, 500 million smart meters had been deployed in Chinese households, while the other parts of the world continue to slowly increase the deployment of smart meters [13].

Another method for energy data gathering is interfaces, connecting the devices to a system. Consumers can input data manually into a website or application, via a computer or a phone for example, that then

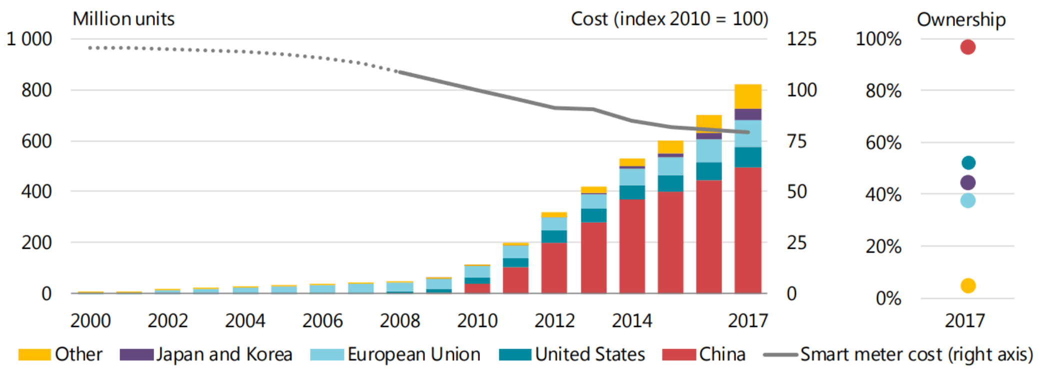

Figure 2.4: Smart meter deployment, with the representation of cost and penetration, from 2000 to 2017 [13].

uploads the data automatically to the network.

As anticipated, all this data needs to be stored somewhere. Distributed ledgers are digital records of data, saved securely, that use independent computers (nodes) to share and synchronize records, alternatively to the traditional ledger that saves all data in a centralized manner. These types of ledgers provide transparency and security, since it is close to impossible to tamper with the data. One of the most known distributed ledger technologies is the blockchain.

After the data is collected, data analysis technologies can combine those large volumes of data and analyze them to construct some kind of instruction or guidance for energy-related processes. Without data analysis, there would only be a large amount of data stored, without any relevance to it. The analysis is what gives meaning to the data, being a crucial step toward energy efficiency. Data analysis can be done through algorithms, simulation software, and even artificial intelligence.

Finally, for this data analysis to have an actual power over energy efficiency, it must trigger an action in the physical environment that will reduce the overall consumption of energy. Thus, it requires a connection between the digital and the physical world. The response to the data analysis can be done automatically, through network communications, or even manually, via actuators and/or human actions.

The actions prompted by the digitalization in the energy sector bring real rewards, but it is still uncertain its impacts on energy demand and efficiency.

The digital devices used in this transition are expanding the energy demand, but they are not considered to be high-energy consumption devices. Some studies suggest that the benefits of digitalization could be nearly ten times greater than any negative impact (Global e-sustainability Initiative and Accenture, 2015 [15]), but still more evidence is needed [13].

In the building sector, both residential and commercial buildings, if there is an increased digitalization of 11 billions connected devices till 2040, it is estimated that it will lead up to 10% less consumption of energy, accumulating energy savings of 234 EJ. Especially in residential buildings, digitalization can save energy if several smart devices are installed and connected to a Home Management System (HMS), that optimizes energy efficiency, leading to a 30% in energy savings [13].

The possible benefits of energy-related digitalization in residential buildings are stated in Figure 2.1.

Technology	Description	Possible benefits
Smart thermostat	Heating and cooling controlled remotely (or automatically) and temperature adjusted according to preferences or sensor inputs.	5-20% less energy use for heating or cooling.
Smart zoning	Individual rooms or zones heated or cooled to specific temperatures at specific times.	10% less energy use for heating or cooling.
Smart window control	Controls amount of light let through and can block heat or cold.	10-20% less energy use for heating or cooling.
Smart lighting (including occupancy control)	Remote control of lighting, automation, adjustment to occupancy.	1-10% less home energy use; 30-40% less lighting energy use.
Smart plugs	Turns unconnected products into connected devices.	1-5% less home energy use.
Home energy management system	Enhances control and automation of energy-using appliances and equipment.	8-20% less home energy use.
Extensive system of management and automation optimised for energy efficiency	Combination of technologies providing measurement, monitoring, dynamic benchmarking, information displays, management, control, automation, zoning, occupancy systems, maintenance management, etc.	30% less home energy use.
Smart district heating	Artificial intelligence combined with sensors to optimise district heating use in apartment blocks.	~10% less apartment block energy use. ~20% more apartment block peak energy savings

Sources: Adapted from 4E TCP EDNA (2018), Intelligent Efficiency: A Case Study of Barriers and Solutions; Leanheat (2019), More Efficient Energy Systems with Artificial Intelligence Controlled Buildings.
Note: Since multiple technologies can address the same energy end-use, the achievable combined savings potential is less than the sum of the potential of different technologies, however additional savings and benefits can be achieved by intelligently co-ordinating technologies.

Table 2.1: Residential buildings: possible benefits of digital technology [13].

Governments, utilities and manufacturers are putting more emphasis on smart grid technologies and digitalization. With this conversion, the deployment of smart appliances is expected to double from 2020-2021 and the number of smart lighting devices is approaching 1 billion [16]. However, security and privacy threats begin to have a more decisive role as smart grids and digital communities start to become more relevant. To ensure that these systems remain secure, new frameworks are needed to handle higher shares of renewables, increased interconnected devices and the changing demand patterns [17].

Thus, the new energy systems need to be more electrified, efficient and interconnected.

2.1.4 Energy Market

Energy markets deal specifically with the trade and supply of energy. Inside this market, the trade can refer not only to electricity, but also sources of energy like oil, petrol and natural gas.

From now on, the focus will be the electricity market. There are two types of markets: the regulated electricity market, and the deregulated electricity market.

The regulated electricity market consists of utilities that manage the system's infrastructure, from the generation of electricity to the meters in the households. These companies own all the infrastructures where the electricity is produced, the transmission lines that bring the electricity to the consumers and the personal equipment each client uses. In Europe, these companies can also manage governmental infrastructures under a license. In this market, the companies sell their electricity directly to the customers, making it a monopoly. Even so, in some countries, there are limitations to these markets, policies in place to regulate the prices, to assure that these monopolies do not overcharge their customers.

The regulated electricity market was the only one to exist, up until the 1970s, restricting the options available to the people. Then came the deregulated electricity market, presenting consumers with a choice, and the prospect to decrease energy prices.

The deregulated electricity market, also known as the free market, allows for the entrance of competitors to buy and sell electricity by permitting participants to invest in power plants and transmission lines. The companies that generate electricity sell their wholesale electricity to retail suppliers. These then set the prices for the consumers, allowing for a competitive market. Consumers can compare the prices and services of different supply companies, often called Energy Service Companies (ESCOs). These companies can also provide different contract structures: a fixed-price contract, where the commodity is sold at a fixed price, independent of the market; an indexed contract, where the price paid is linked to the evolution of market prices (SPOT or forward); and a hybrid contract, a combination between a fixed and index price.

While the regulated electricity market is a monopoly, the free market allows consumers to have a choice on who they are going to buy their electricity from. There is also a bigger availability of renewable sources and green price programs, allowing the customer to specify the amount of electricity to purchase from renewable energy resources.

Within the electricity market, there are three types of markets:

- the day-ahead market, where participants buy or sell electricity one day before the operating day;
- the intraday market, where participants can trade continuously, 24 hours a day, with delivery on the same day (through auctions or continuous trading);
- the forward market, where contracts with delivery and withdrawal obligation are traded, like a bilateral contract, that allows both parties to negotiate the electricity prices and both can share the risks, or a fixed contract.

Right now, the energy market is still deeply dependent on other markets with great volatility, such as the oil, petrol, and electricity markets. Global prices are constantly changing for many reasons, mostly connected to the supply-demand problem: as supply levels decrease, the demand for energy increases, making the market's price fluctuate. Therefore, energy bills usually become erratic unless you have a fixed contract.

2.1.5 Policy Framework

Government policy and access to financial instruments have to follow the energy market in this effort to develop sustainable energy systems. The market must provide the right incentives for consumers to become more active and contribute to keeping the energy system stable.

To address these issues, the European Union (EU), adopted in 2019, a legislative framework that will help accelerate the clean energy transition, the *Clean energy for all Europeans package*, with eight pieces of legislation that will help adapt EU market rules to new market realities [18]. All EU countries, including Portugal, would have 2 years to turn the 8 acts in the *Clean energy for all Europeans package* into national law. The 8 directives of this package are described below [19]:

- Energy performance in buildings - buildings are an important part of carbon emissions in the EU. Since they are responsible for around 40% of energy consumption, decreasing the consumption using energy efficiency measures will help to achieve the goals faster.

- Renewable energy - the EU has set out a laborious target of 32% for renewable energy sources in the energy mix by 2030.

- Energy efficiency - like in the buildings sector, energy efficiency continues to be a central part of the new energy market, as it proves to be the easiest and fastest way of reducing carbon emissions, while also having a major financial advantage to consumers. The EU set the mark of increasing energy efficiency by at least 32.5% over the current level in all sectors, by 2030.

- Governance of the energy union - this act plans to reform the EU energy system, transforming it into a more prosperous governance of the energy system. Each country must create and implement a 10-year National Energy and Climate Plans (NEPCs), to achieve the set targets on all 5-dimensions of the energy union [20]: security, solidarity and trust; a fully integrated internal energy market; energy efficiency; climate action and decarbonising the economy; and research, innovation and competitiveness.

- Electricity directive - reorganize the EU electricity market, making it more flexible and easier to integrate a greater share of renewable energy sources, adapted to the new realities of the energy market.

- Risk preparedness - proposes rules for the cooperation between EU countries with the intention of preventing, preparing and managing electricity crises. Each country must prepare a risk-preparedness plan, based on its own electricity crisis scenarios.

- ACER – Agency for the Cooperation of Energy Regulators - created as a decentralized agency to ensure the implementation of the EU's energy policy objectives in all EU countries.

Most of these directives entered into force in December 2018. These new directives will bring substantial benefits to consumers, the economy and the environment. A new electricity market design will therefore help to achieve the goals set out in the *European Green Deal Search*, and contribute to the creation of jobs and economic growth [21]. A wide range of initiatives has been taken to make consumers an active part of the clean energy transition and help them save more money and energy. These will make the EU a major contributor to the achievement of carbon neutrality by 2050.

Portugal has in place the National Energy and Climate Plan 2021-2030, NEPC 2030 [22], where the objectives, policies and measures to achieve a carbon neutrality economy are laid out. Portugal has strong arguments for continuing to be an example of the energy transition, as the energy policies established have made it one of the top renewable producers in Europe. Data collected shows that renewables and natural gas assumed a bigger importance in the Portuguese energy mix, while fossil fuels, like coal, follow the opposite trend [23]. For example, coal is no longer used since September 2021.

In the NEPC 2030, Portugal defined the targets that will contribute to carbon neutrality, focusing on reducing carbon emissions, energy efficiency, reinforcing usage of renewable energy and reducing energy dependence, moving towards a more sustainable ecosystem. The targets set out are presented in Figure 2.2, which demonstrates that Portugal is on track to meet its 2030 targets.

		2019	2020	2030
Renewable energy share	**Gross final energy consumption**	30.6%	31%	47%
	– Electricity	54%	60%	80%
	– Heating and cooling	42%	41 %	49 %
	– Transport	9%	10%	20%
Energy efficiency	**Maximum primary energy consumption**	22.1 Mtoe (925 PJ)	22.5 Mtoe (942 PJ)	21.5 Mtoe (900 PJ)
	Maximum final energy consumption	17.1 Mtoe (716 PJ)	17.4 Mtoe (728 PJ)	14.9 Mtoe (624 PJ)
Energy import dependency	**External energy dependency**	74.2%	No target	65%
Cross-border electrical interconnection capacity*		10.4 %	10%	15%
Total GHG emissions	**CO_2-eq reduction versus 2005 (excluding LULUCF)**	(2018) -21%	-18% to -23%	-45% to -55%
Total non-ETS GHG emissions (with respect to 2005)		-14.5%	+1%	-17%

* Target based on the ratio of the annual average of commercial interconnection import capacity and generation capacity.

Notes: GHG = greenhouse gas. ETS = Emissions Trading System. CO_2-eq = carbon dioxide equivalent. LULUCF = land use, land-use change and forestry. Mtoe = million tonne of oil equivalent. PJ = petajoule.

Sources: Eurostat (2021a), *Energy from Renewable Sources*, https://ec.europa.eu/eurostat/web/energy/data/shares; Eurostat (2021b), *Complete Energy Balances*, https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=nrg_bal_c; Eurostat (2021c), "Greenhouse gas emissions in ESD sectors", https://ec.europa.eu/eurostat/databrowser/view/T2020_35__custom_822590/default/table?lang=en; UNFCCC (2020), *GHG Total Without LULUCF* (database),https://di.unfccc.int/time_series.

Table 2.2: Portugal's 2020 and 2030 energy sector targets and 2019 status, detailed in NEPC 2030 [22].

Portugal also strategized to promote the creation and development of energy communities, establishing a new decree of law (nº 15/2022), where the rules for energy communities are described. This new legislation establishes the organization and functioning of the National Electric System, transposing older legislation.

2.2 Smart Contracts

Smart contracts are programs that run when predetermined conditions related to a contract between two agents are met [24]. They are typically used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without any intermediary's involvement. These contracts are usually associated with cryptocurrencies or tokens, as a way to trade solely online.

In the case of an energy community, to trade energy in an efficient and fast way, protocols that describe the conditions of the transactions, like the price of energy at that time, must be integrated with information from the network (e.g., smart meter data that describes the amount of energy that was generated, bought or sold) to conduct the energy transaction. These protocols are described in Smart Contracts (SCs), which allow participants to trade based on their own preferences and all the available information.

In general, SCs have the following features [25]:

- pre-written logic rules (execution relies on the independent action of the parties' predefined set of rules);
- stored and replicated on a distributed (control) database;
- executed/run by a network of computers (at least the two agents - buyer and seller - in the contract);
- can result in database updates (e.g. payments).

Smart contracts work by making use of simple "if/when…then…" conditions that are written into software code. When those conditions are met and verified in a piece of information written in the network, the computers execute the action detailed in the contract (see an example in Figure 2.5). In a SC, there can be as many procedures as needed to satisfy the participants' conditions [24]. After the program associated with the activated condition has come to an end, and the final transaction is completed, the system will be updated. Once the overall process has ended, the transaction cannot be changed, and only parties who have permission can see the results [24].

The rules of the smart contract should include every possible outcome. To establish these terms, the framework must be already designed, how transactions and their data are represented. The set of rules that govern those transactions must be agreed upon beforehand, and there must be protocols in place to resolve disputes. Smart contracts can also use incentives or penalties, as a way to drive people to act in good faith.

One can say that the primary objective of this type of contract is the nonnecessity of trusted intermediators. It allows P2P interactions (Peer-to-Peer) to become more accessible and can save time and money to all parties involved.

Below, significant advantages for the use of SCs are listed [24]:

- No need for trusted third parties - the contract is executed inside the network, without the need for any intermediator, which also results in cost reductions.
- Speed - once the conditions stated in the SC are met, the contract is executed immediately.
- Efficiency and accuracy - since SCs are digital, there are no paperwork to process and no time spent reconciling errors that are oftentimes made manually.

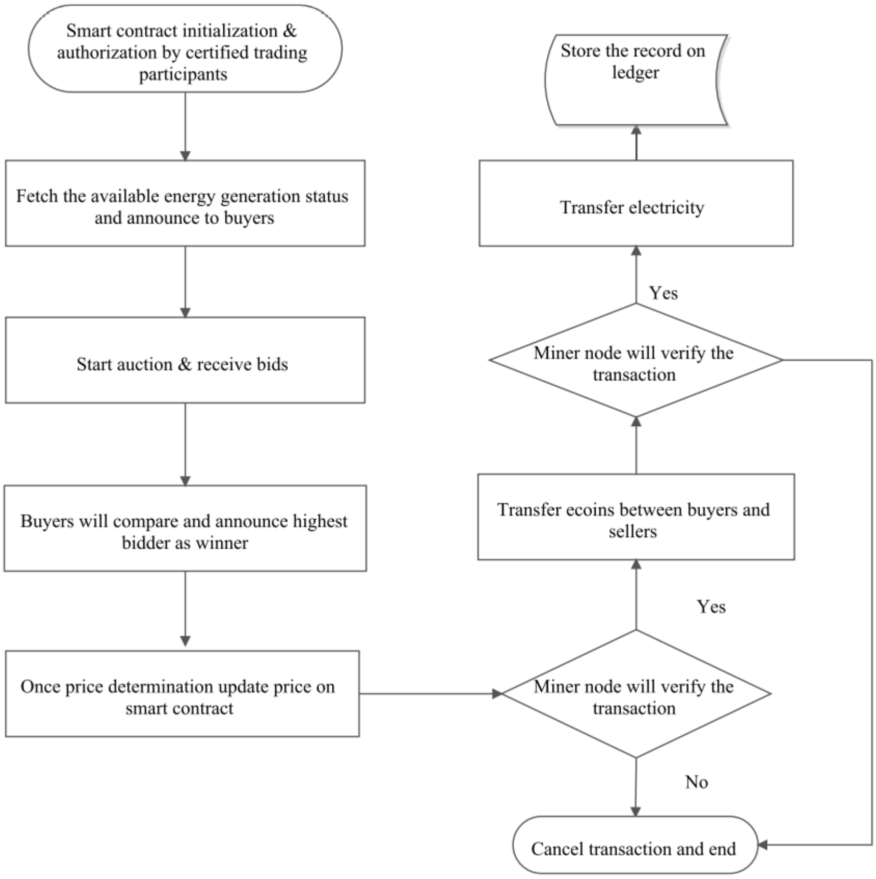

Figure 2.5: An example for an algorithm of smart contract for P2P energy trading [26].

- Trust and transparency - the terms of these contracts are known by all parties involved, and their transparent nature removes the possibility of manipulation or error.
- Security - SCs have the highest level of data encryption available, making them very hard to hack. Moreover, when connected to a distributed ledger, to tamper with it, one needs to alter the entire chain to change a single piece of information.
- Storage and backup - the records are also kept in digital format, making them easier to store and in the possible event of data loss, they are easily retrievable.
- Savings - SCs eliminate the need for any other entities' involvement, such as lawyers, banks, etc.

Compared to traditional contracts, where terms are agreed upon between parties involved and they must rely entirely on the honor of the intermediator, SCs bring significant improvements. In Figure 2.6, the main differences between the traditional contracts and SCs are declared.

One must also bear in mind that a smart contract is not necessarily a valid contract in the eyes of the law. There are several discussions around this topic, as some claim that SCs cannot be considered legal agreements, they only perform the actions set out by other legal agreements. They are technological means for the automation of payment obligations, obligations consisting in the transfer of tokens. Moreover, others state that the nature of programming languages can impact the legal validity of SCs [28]. These contracts work very well when the conditions are quantifiable and objective. A SC's binary nature will

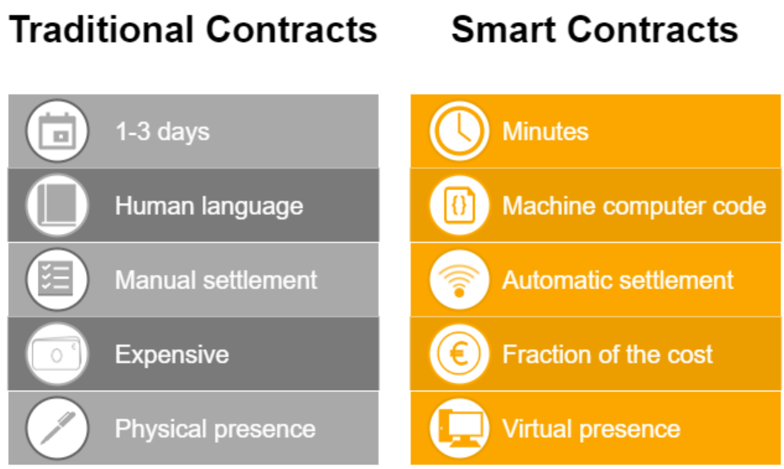

Figure 2.6: Key differences between traditional contracts and smart contracts [27].

slip when confronted with subjective terms or conditions, like when "good faith" or "reasonable" are applied. Nevertheless, smart legal contracts exist, SCs that are legally enforceable by the law, since they have all the elements of a legal contract.

Still, such as traditional contracts have disadvantages, SCs can also bring other complications. Given its tamper-proof essence, if any error is made, it is almost impossible to directly correct it. And since these contracts are executed in a digital network, security becomes an important issue.

Notwithstanding, the characteristics of SCs make them ideal for P2P energy trading in local communities. By setting tamper-proof execution of computer code, the records in the database cannot be modified unilaterally by either the parties involved [25]. And when we connect smart contracts with smart meter data, energy efficiency may be enhanced by enabling the automatic measurement of energy generated or consumed and automatically adjusting demand and supply, when described in the terms of the contract.

Smart contracts can be implemented using blockchain technology, a distributed ledger. For this reason, smart contracts are becoming crucial to decentralized applications and blockchain development in general. More and more organizations that use blockchain for business are developing templates, web interfaces, and other online tools that simplify the implementation of smart contracts [29].

2.3 Blockchain

A blockchain, popularized in 2008, is a shared ledger (book-of-records) that facilitates the process of recording transactions and tracking assets in a network [29]. As the name indicates, it consists of blocks, that contain information, and are binded to each other, through a chain. In this particular network, the asset is energy, but it can also have other tokens associated with it, something that is exchanged for commodities, in order to make transactions simpler.

The main key concepts behind a blockchain are [30]:

- Distributed ledger technology (DLT): all participants in the network can have access to the ledger and its records of transactions. With this digital ledger, the information is recorded only once, eliminating duplication inside the system, common in traditional ledgers.

- Immutable records: No participant can tamper with a transaction after it has been added to the blockchain. If, for some reason, a record has an error, a new transaction can be added to the blockchain in order to reverse the error, and both transactions are visible to participants.
- Smart contracts: to speed up the system, smart contracts can be stored on the blockchain and executed automatically.
- Permissions: permissions are what protect the network, it ensures that the transactions are authenticated and verifiable. This also grants data protection and privacy.
- Consensus: through consensus algorithms, the network can verify the transactions. There are several consensus mechanisms, discussed ahead.

Blockchains are immutable, decentralized, and saved across several networks. The information is written in blocks, each connected to the one before and after it, via cryptography. Once the writing block process is complete, it becomes almost impossible to tamper with it. The transactions are recorded only once and are visible to all participants in the network. As a result, trust, accountability, and transparency can be expected.

2.3.1 Structure of the blockchain

The layered structure of the blockchain architecture [31], represented in Figure 2.7, is:

1. Hardware / Infrastructure layer: comprises of hardware, protocols, connections and other components essential to the foundation of a blockchain. This layer can be thought of as layer 0, the network of the blockchain.
2. Data layer: this layer contains the chain of blocks that store the transactions, a series of hashed blocks carrying transactional records. The blockchain begins with the genesis block (block 0), after that, every new block added to the blockchain is linked to the genesis block through an iterative process.
3. Network layer: also known as the propagation layer, it is composed of all nodes that belong to the network, facilitating inter-node communication. It controls node identification, block production and adding.
4. Consensus layer: fundamental to blockchain functionality, it is responsible for the validation of the transactions that go through the network. Each transaction is processed by multiple nodes, all of which must arrive at the same conclusion (e.g. consensus). This layer is what makes the blockchain decentralized, since no node has full control over any data.
5. Application and presentation layer: it consists of the programs that the end-users take advantage of, to establish blockchain network communication, such as smart contracts and decentralized Applications, dApps.

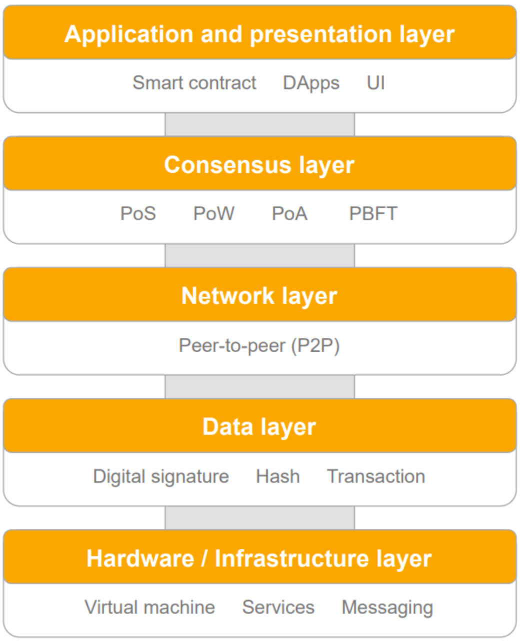

Figure 2.7: Layered structure of the blockchain architecture [31].

As for the structure of the blocks in a blockchain (see Figure 2.8), it is constituted by[32]:

- Block identifier - block's address, hash and height. The hash is what differentiates a block from another, it is generated by SHA256 algorithm that produces unique and unrepeatable identifiers from given information. The height represents the number of blocks between the current block and the genesis block (block 0).
- Block header - contains information: timestamp, previous block hash, nonce (number only used once to identify the block) and merkle root (method to confirm that the data in the block has not been altered).
- Transaction - list of the transactions in that block.

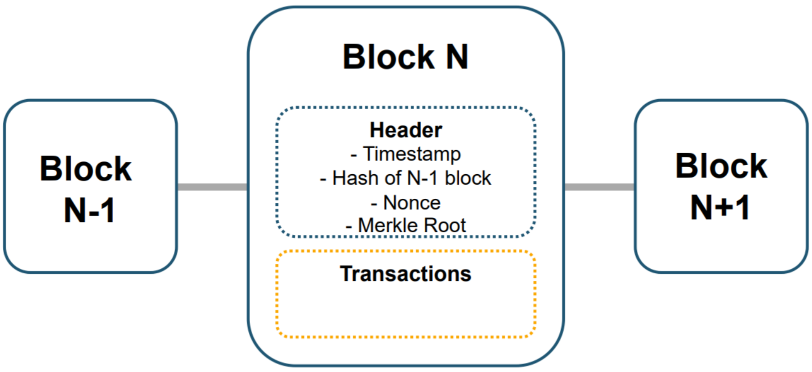

Figure 2.8: Basic structure of a block inside the blockchain [32].

A blockchain can also variate in its degree of decentralization. There are mainly 3 types of blockchain technology: a public blockchain, where anyone can join and participate; a consortium blockchain, where multiple organizations share the responsibility of maintenance; and a private blockchain, where one organization governs the network. Table 2.3 presents the fundamental characteristics of each type of blockchain and some examples of frameworks.

Characteristics	Public	Consortium	Private
Degree of centralization	Decentralized	Weak centralized	Centralized
Motivation from participants	Necessary	Selective	Not necessary
Consensus determination	All agents	Selected nodes	One organization
Read permission	Public	Public or restricted	Public or restricted
Immutability	Near impossible to tamper	Could be tampered	Could be tampered
Efficiency	Low	High	High
Consensus process	Permissionless	Permissioned	Permissioned
Framework	Bitcoin, Ethereum	RIPPLE, LIP-chain	Byte

Table 2.3: Comparison between types of blockchains [33, 34].

Each type of blockchain has its own advantages and disadvantages. Some applications require that an organization has to be responsible for the network, while others need full transparency and decentralization.

2.3.2 Consensus Algorithms

For a blockchain to work properly, all blocks must be checked to validate the information. The blockchain network must be able to work efficiently, even in the presence of dishonest information. For this purpose, consensus algorithms are created in decentralized systems, a common agreement between nodes, that ensures validity and prevents manipulation [26]. The consensus mechanism also assures synchronization between different blockchain nodes and a shared public ledger, and enables network nodes to reach a disputable free agreement without any third-party.

The most common types of consensus algorithms are:

- Proof-of-Work (PoW): When using PoW, the node with the highest computational power usually mines the block, given that is less likely that it will attack the network. Miners need to compete with each other to add a new block to the blockchain, by solving a problem through trial and error method. This requires great computational effort. When a block is validated, the block returns to the blockchain. Other nodes will then verify if the transactions are valid or not. The miners who perform the work are compensated with a financial reward. Bitcoin is currently using this algorithm to mine transactions.

- Proof-of-Stake (PoS): In the case of PoS, the algorithm requires a stake. Owners can offer their currency as collateral to become validators. To attack the network, one would need to own the majority of stakes, that is 51%. This algorithm surfaced because of the criticisms done to PoW, where a large amount of energy and resources are used for mining. PoS replaces the computational effort with staked currency, where the chance of successfully mining the block is proportional to the staked values of the validator. Validators also receive rewards when confirming that a block is indeed correct. This approach can potentially result in a faster mining process with fewer resources, and that is why PoS is becoming more popular than PoW. However, this system can become quite a monopoly, since the richest users of the network are likely to be dominant in this consensus.

- Proof-of-Authority (PoA): Unlike PoS, PoA uses the identity as a stake, it only allows approved nodes to validate the information. In this case, the authority must remain uncompromised.

- Practical Byzantine Fault Tolerance: PBFT provides a Byzantine fault tolerance algorithm, that can be achieved if the loyal nodes have a majority agreement on their strategy.

All consensus algorithms have different strengths and weaknesses. A more detailed description and a comparison between the algorithms stated above is available in [26, 35].

In Figure 2.9, the process for validating a transaction is explained. First, a transaction must be requested and authenticated. Right after, a block that includes that transaction is created and sent to the network, so that selected nodes can validate the transaction. The validation depends on the consensus algorithm selected. After the validation is completed, the nodes receive an incentive to continue and confirm transactions. Then, the block is added to the blockchain, the updated blockchain is distributed across the network and finally the transaction is completed.

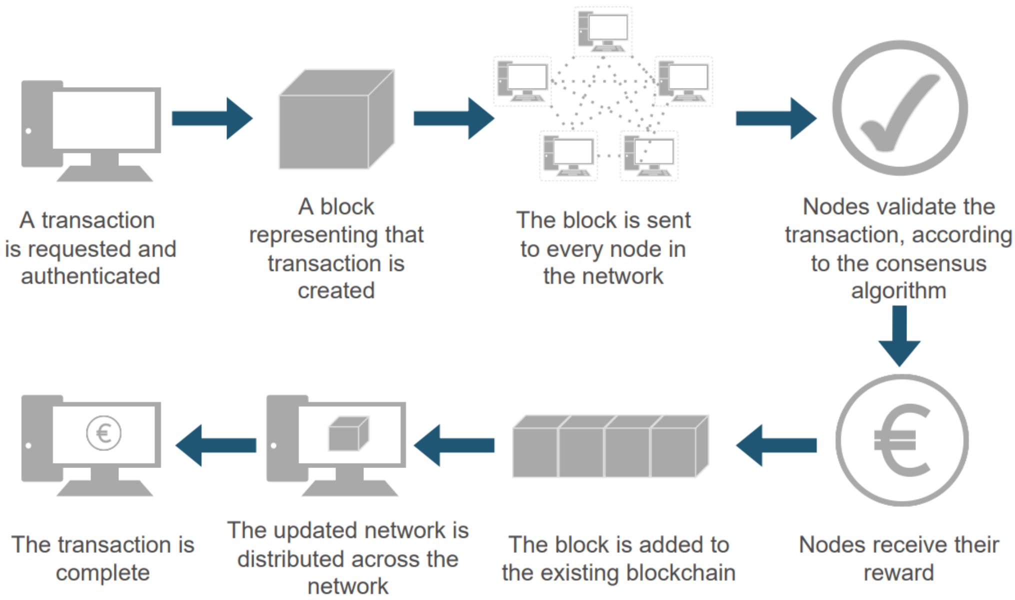

Figure 2.9: General process to get a transaction into the blockchain [36].

Smart contracts using blockchain technology are an unprecedented opportunity to perform the transitions in the new energy system, by promoting decentralized energy trading, highly scalable and secure.

2.4 Blockchain-based Smart Contracts

Since the popularization of blockchains, especially Ethereum, the term smart contract has been more explicitly practiced collectively with a blockchain, since it describes the computation that takes place and operates within a blockchain.

United with blockchain technology, the execution of a SC is controlled and audited by the platform, and not by any program outside the network. The SC is directly connected to the blockchain, making it visible to all nodes of the blockchain.

To build such an application, the best practice is to follow the guidelines presented in Figure 2.10.

Figure 2.10: Guidelines to build blockchain-based smart contracts [37].

These platforms can add some risks. If the SC includes vagueness or 'easy-but-insecure' construct, the platform will not be successful. Additionally, a digital tool like a blockchain comes with some possibilities, such as cyberattacks and fraud, which can be incredibly difficult to resolve.

Aside from this, blockchain-base smart contracts are still a viable, if not captivating, solution for energy trading systems, especially in P2P trading, where there is a need for:

1. Decentralization - crucial for P2P trading, where there is not a trusted third-party.
2. Security - all the information is stored in blocks, which are close to impossible to tamper with.
3. Traceability - every block is unique and traceable.
4. Transparency - the data saved in the blockchain (information and transactions) are visible to participants in the community.
5. Automation - the system is run online and can be automatized to need little human intervention.

Blockchain-based solutions can be ideal for an energy trading structure because they provide immediate, shared and completely transparent information stored on an immutable ledger. The faster we receive information and the more accurate it is, the more efficient the network can be. The blockchain can track, within the network, orders, transactions, payments, production, consumption data and much more. And since the participants share a common goal - a more sustainable energy system - new opportunities for the use of renewable energy and efficient consumption are within reach.

In the next section, the current developments in this field are discussed.

2.4.1 Current Developments

The application of blockchain and smart contracts technology in the energy sector has been gaining more and more attention. There has been a considerable development of studies and initiatives about the use of blockchain in the energy sector, predominantly in Peer-to-Peer (P2P) energy trading communities, since blockchain can connect and coordinate a large number of participants in the same area.

Several papers review the state-of-art of blockchain-based smart contracts applied to different energy communities [26, 35, 38, 39]. In this section, some of the current developments of blockchain for energy trading are presented.

In [40], it is proposed a solution for energy trading, based on the blockchain Ethereum and smart contracts, that satisfy the needs of energy trading in smart grids, focusing on the privacy of the participants (consumers, prosumers and retailers). This smart contract is deployed by a third-party entity, the Energy Authority.

The solution proposed in [41] also uses the Ethereum blockchain, where consumers and prosumers in a sustainable local community trade without the need for an outside authority. This design only allows the participation of authorized users, as it is a private blockchain.

The authors in [42] introduce a secure private blockchain-based platform guarantying the privacy of producers and consumers. The consumers' privacy is preserved by changeable public keys of their smart meters, and the negotiation between both parties is conducted off-the-chain.

In [43], it was designed an entirely new framework, SmartChain, inspired by blockchain technology and aiming at low computational complexity, to secure P2P energy trading in a smart grid ecosystem. The framework was evaluated using the parameters such as execution and validation time. The obtained results depict the superiority of SmartChain in contrast to the conventional blockchain process.

In [44], an energy prosumer business model using blockchain technology is developed, that allows various energy sources to be connected to various users and producers. To improve energy efficiency, the energy pattern of users is analyzed. With the method implemented, energy shortages appeared to be resolved, the role of prosumers would be more relevant and the consumers would have more power of choice.

Other types of blockchain-based solutions, such as a P2P blockchain where agents trade bilaterally [45, 46], enabled the implementation of an energy management system for renewable energy [47]. Double-layer energy blockchains [48] have also been studied.

Clearly, blockchain-based frameworks have a distinct interest in small local energy communities, where there is a need for a decentralized structure.

Despite noticeable conveniences connected to the implementation of blockchain technologies, there are still some issues that act as a barrier to their practical application. A main concern is the privacy of every agent integrated in the network. All the information that is written in the blockchain (that can contain consuming habits, personal information, etc.), cannot be removed. Despite the analysis made, this problem still poses a serious issue, especially when legislative norms are taken into consideration [49].

The problem of personal data privacy also applies to public energy communities. Public services have also been at the center of studies in the energy sector. Blockchain-based solutions may be applied to the public sector [50], with the same concerns for privacy, even if part of the data is already public.

At the moment, some projects have been implemented in real communities, such as Grid+ [51], Power Ledger [52], and Brooklyn microgrid [53], developed in P2P energy grids.

The overall conclusion of these studies and real cases is that blockchains and smart contracts provide clear benefits to the energy system, markets and participants.

Chapter 3

Methodology

In this section, the context in which the application will be evaluated is presented. Before starting the implementation, it is necessary to define what type of users can participate in this network, so that the smart contracts include every possible situation that can occur within the community, and in what type of situations the application will be tested. Lastly, three different models for a smart contract applied to energy trading are described.

3.1 Context

For testing purposes, it is considered an energy community with 4 prosumers and one Energy Services Company (ESCO). Only prosumers are considered in this community since considering a consumer would not bring any additional difficulty to this problem. So, in the interest of a highly scalable application, these 4 participants contemplate the 2 types of outcomes for energy trading: participants that want to buy energy and participants that want to sell energy. An ESCO is included in this energy community to supply energy and energy-related equipment when needed, such as the infrastructures to connect all participants and smart devices.

Prosumers will own electricity-generating equipment, such as a photovoltaic system. This equipment also contains a small storage (battery), where the energy that is not currently being used will be stored. Furthermore, there is also the possibility of installing a larger energy storage, as to not become so dependent on the amount of electricity that the battery can hold. The electricity that is produced can be used around the house, to supply energy to electronic devices such as refrigeration, kitchen utensils, and even charge an electric car. When their produced energy surpasses the energy demand and the storage is full, the electricity must be put back into the utility grid, or sold to someone.

The participants in the energy community can buy or sell energy according to their own preferences. They can choose to buy electricity only for instant consumption, or choose to trade according to their energy storage.

All participants, including the ESCO, are connected to each other via the electricity grid and the application. Furthermore, each participant establishes communication with the blockchain technology individually (refer to Figure 3.1, for the representation of the community).

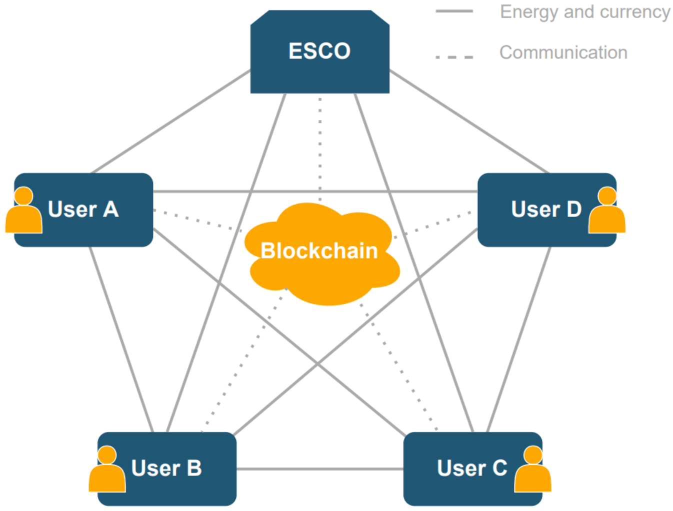

Figure 3.1: Design of the energy community.

The participants can also install smart meters inside their house, devices that communicate energy-related data, namely consumption and production data, to the user for finer system monitoring. This allows the user to optimize the consumption behavior and get access to production and/or consumption patterns, fitting for an energy trading community. At least, one smart meter must be connected to the electricity grid that goes inside each house, to account for the flow and ensure the transfer of electricity within the community.

To facilitate the process of trading, there is also the possibility to invest in a Home Management System (HMS). These systems can manage the production, consumption and storage of the user. It receives the data from the smart meters installed around the house, and with that data, it can predict production and consumption data for the future. With those predictions, the system can automatically make decisions for the owner, including when to buy or sell energy, and the amount of energy.

Both systems allow the user to trade energy more efficiently, based on decisions founded on the data associated with each particular situation.

However, these valuable systems are not within the scope of this project. If the user has a HMS installed, it could communicate how much energy is necessary to sell or buy energy, and even insert the data directly into the application, if a connection is established, without the need for the user's approval. The application will consider that the data of the electricity to be sold or bought for each user is already in a secure database.

As expected, the ESCO inside the community can provide energy for all participants, and also buy energy from the community. The community can choose what type of contract with the ESCO they want. The ESCO will only buy or sell energy to the community, according to the contract agreed upon beforehand, while also supplying maintenance for the devices installed and the electricity grid, connecting all participants in the community.

As for the four users, their energy-related description is:

- User A: the PV system is not sufficient for the demand, so this user will only buy energy from the community. With a large storage, this user can buy large amounts of energy at once.
- User B: the PV system covers a big part of the demand, however, this user's storage is not enough to accommodate the energy provided by the PV system. So, this user will sell some of its energy to the community, and sometimes he will need to buy energy.
- User C: the PV covers a small portion of the demand, so he will mostly buy energy from the community. Sometimes this user will need to sell his energy due to the lack of storage.
- User D: the PV system covers all demand and there is enough storage, so this user will sell the extra energy to the community.

For each of the users, the balance is defined in one moment in time. This balance determines if the user wishes to sell or buy energy, decided by the '+' or '-' respectively, and the amount of energy.

The production and demand patterns are not included in this thesis. However, the values of the balances were selected considering possible production and consumption data for the descriptions given above for each user.

In Table 3.1, the balance of all users is presented. These are the values that will be used when testing the application.

User	Balance (MWh)
A	- 0.007
B	+ 0.005
C	- 0.003
D	+ 0.006
TOTAL	+ 0.001

Table 3.1: Energy balance of each user, in one moment in time.

The interaction between the community and the application will be evaluated in 4 different scenarios, to compare the influence of the time and electricity market prices:

- During the day with low electricity prices,
- During the day with high electricity prices,
- During the night with low electricity prices,
- During the night with high electricity prices.

The electricity prices can be seen in OMIE, the nominated electricity market operator for managing the Iberian Peninsula's day-ahead and intraday electricity markets. Inside this electricity market operator, there is what is known as the SPOT market.

A SPOT market is where a commodity or other assets are traded for immediate (or very near-term) delivery. The official transfer of funds between the two parties may take time, but both agree to trade the commodity the moment the transaction is made. The current price of the assets is called the spot price, and it is the price at which the assets can be sold or bought immediately.

The prices for the SPOT market can be seen on the OMIE website [54]. This market displays the electricity price hourly.

The smart contract with the ESCO will define its prices according to the SPOT market for electricity. To test the application, real data from the SPOT market is used. The prices for the 4 different scenarios are selected from the month of September 2022. For the days where the electricity price is low, it is also considered a low average price for that day, once it will also be a part of the smart contract. The electricity prices to be considered are presented in Table 3.2.

Description	Price (€/MWh)	Average price (€/MWh)
Day – low prices	60.50	100.16
Day – high prices	221.29	175.42
Night – low prices	75.64	119.77
Night – high prices	248.80	193.04

Table 3.2: Electricity prices for the four different scenarios to be tested.

3.2 Smart Contracts

Since energy trading inside the community can include many options, such as trading only with the ESCO and trading amongst peers, it was decided that the best approach to this problem was to divide it in steps. As a result, three different energy trading smart contracts were tested, each one more complex than the other.

In the following sections, each one of the models, and how the energy can be traded within the community, is explained. All models permit the person to use smart meters and HMS to get information on the amount of energy to trade, making conscious decisions based on real-time information.

3.2.1 First Smart Contract

The first design (Figure 3.2) only considers trading between users and the ESCO. Each user can buy energy directly from the ESCO or sell energy directly to the ESCO. There is no P2P trading in this model. The price at which the electricity will be sold is established in a generic smart contract. Each user can define at what time the trading occurs and the amount of energy they wish to trade.

This model serves mainly the purpose of testing out the smart contract that will be used for all interactions with the ESCO in the next designs. There is no distinct reason to implement this contract in a blockchain-based application.

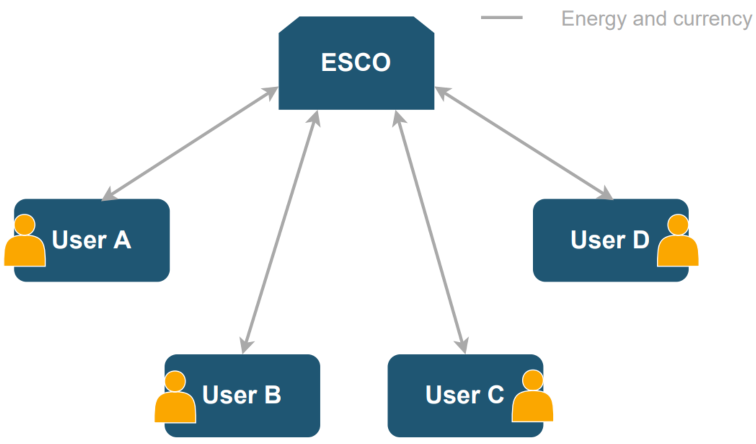

Figure 3.2: Architecture of the first smart contract for energy trading.

3.2.2 Second Smart Contract

In the next model (Figure 3.3), P2P trading was added, but in very a straight-forward way. The energy can be traded between all users in the community. Yet, all trading still follows the same price-setting rules as before, the prices that were used for the ESCO trading in the previous model.

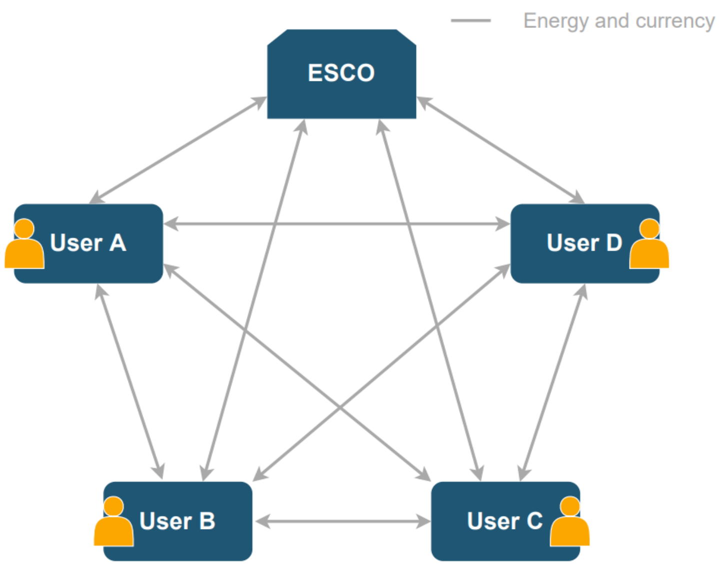

Figure 3.3: Architecture of the second smart contract for energy trading.

Here, all participants can input the energy they want to buy or sell. The application will prioritize the trade between peers and users that requested a trade first. As a last resort, the system will go to the ESCO to finalize all pending trades.

3.2.3 Third Smart Contract

The third and last smart contract follows the same architecture as the previous model (Figure 3.3). There will be P2P trading, but the trade will be done through auctions. It will be used a first-price sealed-bid auction, also known as a blind auction, and the program will function as the auctioneer. In this type of auction, all bidders can submit sealed bids. In this way, no bidder can know the bid of other participants. The highest bidder wins and pays the exact price that he bid during the auction, while the losers don't have to pay anything.

The users can start auctions when they have energy to sell. They define the amount of energy to sell, and other users can place bids anonymously to buy the entirety of the energy auctioned. After the auction ends, there is a winner, and the program will then proceed with the trading among them. In case there are equal bids in one auction, the winner will be the participant that bid first. It is also included the option to trade with the ESCO, on their own terms, for users that wish to trade immediately or when there are no auctions available.

Chapter 4

Implementation and Results

In this chapter, it is explained how the solution was implemented. It starts with defining a generic smart contract that will be applied to all interactions regarding the ESCO. Then, the foundation of the blockchain is explained. A web user interface is created to connect all users with the application, and it is also made clear how the user can handle their cryptocurrency.

The selected blockchain technology is the Ethereum blockchain, with ETH (ether) as its cryptocurrency.

The implementation of each smart contract is then described and the results are presented.

4.1 Smart Contract

As explained in section 2.2, a smart contract is a self-performing contract that executes when predetermined conditions are met. The smart contract that will be used for this application will define the terms to trade electricity between the participants, buyer and seller, focusing primarily on the transaction of currency for the amount of energy established.

Smart contracts are written in Solidity language, an object-oriented programming language used for implementing SCs, runned on the Ethereum Virtual Machine.

The smart contract will have the following components (independent of the model):

- Struct TransferStruct – allows for the creation of the transfer data, including the address of the sender and the receiver, the amount of energy requested, the number of ETH transferred and the timestamp. This will define the components that will be added to each block, displayed in the platform.

- Function addToBlockchain – where all the data is added to the blockchain of this smart contract. Inside this function, an event is emitted, i.e., the transfer of ETH is included inside this function.

- Function getAllTransactions – access all transactions done through the smart contract.

- Function getTransactionCount – get the number of transactions done through the smart contract.

After getting the structure of the smart contract, a generic smart contract for the ESCO is planned. The contract will follow the prices of the SPOT market, as stated previously, where the electricity is sold at the price on that specific moment.

The design of the generic smart contract for interactions with the ESCO is the following:

1. The smart contract reads the electricity price for the moment the transaction is meant to begin.
2. If the price is lower than a variable P, then the price that will be paid is the average of the hourly prices for that day.
3. If the price is higher than P, then the price will stay the same.
4. After defining the price, in €/MWh, the SC will calculate the number of ETH to pay for the electricity requested, by multiplying the amount of energy in MWh, the price of the electricity defined in steps 2 or 3 and the exchange rate of € to ETH. It is also applied an energy trading fee of 10% when the user wants to buy energy from the ESCO.
5. The number of ETH to transfer between buyer and seller will then be cut to 8 decimals, to assure that it is a finite number.

The variable P was defined as $P = 150$ €/MWh, which represents the average of the electricity price in the last few months, from June to August of 2022. This type of contract protects the ESCO, in case the electricity prices run much lower than the average. This means that, for the low-price scenarios, the SC will use the daily average for the computation, while for the high-price scenarios, the actual price is used.

The trading fee is only applied when the ESCO sells energy, to make sure that the P2P trading is profitable for all parties, in the next smart contracts.

So, this smart contract needs two external values to complete a transfer: the electricity prices and the exchange rate of € to ETH.

These values can be fetched through an API (Application Programming Interface), an interface that contains a set of functions to help developers access specific parts of data in an application. In the case of a web API, where the data is in a website, the data can be accessed using HTTP protocol.

However, fetching real-world information for a smart contract poses a challenge. The smart contract will be run through a blockchain, where the blockchain will use its nodes to execute the smart contract and arrive at a consensus. Several nodes will execute the action of getting the information from real-world data directly, which could possibly result in nodes having different values, since the nodes will not get the information all at the same time. Consequently, they cannot reach a consensus, since they begin with different information. To solve this problem, is common to use an oracle, a single node that acts as a bridge between the blockchain and the real-world. Instead of the smart contract accessing an API directly, the oracle will fetch the data from the API, and the smart contract can request the data from the oracle node (see Figure 4.1).

For the electricity prices, [56] was used as an API. This API can access the SPOT market prices, just by changing the '*start_date*' and '*end_date*' parameters. For example, if one wishes to access the prices

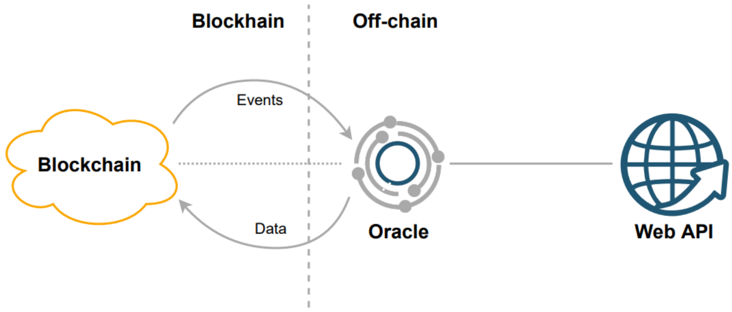

Figure 4.1: Diagram of oracles used in blockchain technology [55].

for September 23rd, the following URL would be used: `https://apidatos.ree.es/es/datos/mercados/precios-mercados-tiempo-real?start_date=2022-09-23T00:00&end_date=2022-09-23T23:59&time_trunc=hour`.

Requesting from this API, the smart contract receives a multi-dimension array, where the hourly prices of the specific day in two markets are saved: the PVPC market and the SPOT market. In this application, only the SPOT prices are relevant. From the SPOT prices, the smart contract can then extract the electricity price for the hour in question and even calculate the daily average.

For the exchange rate of € to ETH, an API was not used, given that the market is always changing and would make it much more difficult to compare the final results if the conversion was changing between tests. For that reason, a fixed number for the exchange rate was used, 0.00065 ETH to 1€, an average of the price in the last few months.

With this information, it is then possible to execute the smart contract and transfer the funds between buyer and seller.

Note that SCs need to be deployed. These deployments will have a cost, directly connected to the complexity of the SC. Moreover, every time the contract is used, that is, anytime a transaction goes through the SC, a fee will be applied to the sender.

There will be three different models for trading within the community. For each of them, there will be a different set of rules of interaction between the users, only the interaction between a user and the ESCO remains unchanged between models. The details of the smart contract applied to each model are stated within their respective section.

4.2 Blockchain

The smart contract above will be run through the Ethereum blockchain. Ethereum is an open-sourced and decentralized blockchain, with ether (ETH) as the cryptocurrency, that powers thousands of decentralized applications, dApps.

This blockchain is commonly used for trading-related applications. And it has updated documentation online, making it accessible to new developers.

4.2.1 Ethereum Proof-of-Stake Algorithm

Ethereum is a technology used for building apps, where users can own assets and transact between them, without being controlled by a central authority. This blockchain is currently using proof-of-stake for the consensus mechanism. Before the Merge in 2022, the Ethereum blockchain was using proof-of-work based consensus protocol. Note that while developing the application, PoW was the consensus algorithm. When the final results were conducted, the Merge had been finalized, and PoS was in place.

The network is composed of nodes, and each node communicates with a small subset of the network – called 'peers'. Whenever a node wants to add a new transaction to the blockchain, it sends a copy of the transaction to its peers, who send a copy to their peers, propagating throughout the network. Then, validators, selected nodes in the network, participate in the consensus mechanism. Validators stake ETH into a smart contract on Ethereum to participate in the system, and this staked ETH acts as collateral that can be destroyed if a validator behaves dishonestly or lazily. They are chosen at random to check if new blocks are valid and can occasionally create and propagate new blocks. After the block has been checked, the validators send a vote, called attestation, in favor of that block across the network. The votes of all validators that checked the same block are then used to determine the validity of the block being proposed [57].

Instead of needing to do intensive computational work (PoW), validators only need to stake their ETH in the network. There is also a system of rewards and penalties to incentivize healthy network behavior.

Validators receive rewards when they make votes that are consistent with the majority of the validators pools and when they propose new blocks. The reward is proportional to the validator's effective balance and inversely proportional to the number of validators on the network. An additional reward is added to incentivize rapid attestations. Block proposers can also increase their reward by including evidence of misbehavior by other validators. These rewards encourage validator's honesty. On the other hand, if a validator does not work or works very slowly, instead of receiving the reward, they have an equal amount of ETH removed from their balance. There is also the possibility of ejection from the network if the validator is behaving against it [57].

Thus, to use an Ethereum application, the user must pay a fee in ETH, a reward given to validators to check and prove that no one is cheating and perform work for the right to propose a block of transactions. The work the validators do, keeps Ethereum secure and free of centralized control.

4.2.2 Ethereum Virtual Machine

The Ethereum Virtual Machine (EVM) [57] is the runtime environment for transaction execution in Ethereum. It includes a stack, memory, gas balance, program counter and persistent storage for all accounts. The smart contracts in this project will be developed in a Testnet EVM that includes all essential components of the EVM. The EVM is isolated from other files and processes on the node's computer to ensure that for a given pre-transaction and transaction, every node produces the same post-transaction state, thereby enabling network consensus.

Gas is a unit within the EVM used in the calculation of the transaction fee, the amount of ETH the sender must pay as an incentive to the validator. Each type of operation performed by the EVM has a specific gas cost, roughly proportional to the number of resources (computation and storage) used to conclude such an operation. The number of incentives the user wishes to pay the miner influence the amount of time that takes the blockchain to conclude the operation [57].

Before 2021, the sender would specify the gas limit and gas price, the maximum amount of gas the sender is willing to use in the transaction, and the amount of ETH the sender wishes to pay to the miner per unit of gas used, respectively. At each transaction, the sender buys the full amount of gas (gas balance): $gas\,limit \times gas\,price\,upfront$, meaning that with every transaction, the sender is also debited the gas balance amount. At the end of the transaction, the sender is refunded for any unused gas.

This leads to fixed-size blocks, which in high demand times, the user would have to wait for the reduction of demand to get included in a block, leading to great waiting periods. The London Upgrade in 2021 introduced variable-size blocks, blocks that would increase or decrease in size depending on the network demand, creating a more effective network.

After the London Upgrade, the system in place nowadays calculates the total fee for the gas balance as: $gas\,used \times (base\,fee + priority\,fee)$, where the base fee is set by the protocol and the priority fee is set by the user as a tip to the validator. Additionally, one can set a maximum fee for the transaction, then the difference between the maximum fee and the actual fee is refunded.

In user's interfaces, gas prices are typically denominated in Gigawei (Gwei), a subunit of ETH equal to 10^{-9} ETH. The higher the gas price, the quicker the transaction is included in the blockchain, given that the validator has a bigger incentive to include that specific transaction in the blockchain.

In Ethereum, there are two types of accounts: user accounts - Externally Owned Accounts (EOA), and contract accounts. Both accounts can receive and send ETH or other tokens and interact with smart contracts, however, a contract cannot start a transaction on its own. User accounts are controlled by public and private keys and can create transactions, while the contract accounts have code and storage associated, sets of functions and variables (at any given time), that execute when it receives a transaction from an EOA. It is controlled by the smart contract logic. The contract address is usually given when a contract is deployed to the Ethereum network and it is where all the contract deployment and interaction fees will go to, while the EOAs are created outside the network [57].

A transaction usually includes the following information [57]:

- Recipient - the receiving address,
- Signature - the sender's address,
- Nonce - transaction identifier,
- Value - the amount of ETH to transfer,
- Data - any additional data to include,
- Gas Limit - the maximum amount of gas units to be used in the transaction,

- Maximum priority fee per gas - the maximum amount of gas to be included as the tip,
- Maximum fee per gas - the maximum amount of gas the user is willing to pay for the transaction.

As anticipated, the receiver and sender addresses, and the value are identified in the SC. Only the gas limit was defined at 21 000 Gwei, a standard value for ETH transfers, while all other values for the gas are set to default. And the following data was added to each block:

- Timestamp - time of the transaction,
- Energy requested - the amount of energy to be transferred between sender and receiver.

4.3 Web Interface

To make the application user-friendly and as real as possible for testing, a website was created as the UI. For this, Web3.0 is used – decentralized internet based on public blockchains, being built, operated and owned by its users.

Web3 uses blockchains, cryptocurrencies and NFTs (non-fungible tokens) to give users ownership. A few key principles to better understand Web3 are:

- Decentralization: ownership is distributed amongst its builders and users,
- Permissionless: every user has equal access to participate,
- Native payments: it uses cryptocurrency to transact money online,
- Trustless: it operates using incentives, without having to trust a third party.

With Web3, it is possible to create a decentralized app that runs on the Ethereum blockchain.

A collection of libraries in JavaScript, web3.js [58], is used with the main purpose of interacting with the Ethereum blockchain. This library facilitates the development of websites that connect clients with the blockchain, by allowing the users to create smart contracts and perform transactions.

Several development tools were used to create the application, and connect it to the Ethereum blockchain to run a test. The combination Vite + React is used to create the UI. React [59] is a JavaScript library to make interactive UIs, that are fast, beginner-friendly, and will efficiently render only the components that were updated. Vite [60] can install, build and start the React app in a few seconds. For the design of the website, TailwindCSS [61] is added, a cascading Style Sheets (CSS) framework to help build a modern website, describing the presentation of the HTML, faster and easily.

Finally, to connect the UI to the Ethereum blockchain, HardHat and Alchemy are used. HardHat [62] is an Ethereum development environment that allows to run solidity locally, i.e., test smart contracts before deploying them. This framework will give the contract address to which the smart contract will be deployed to for testing. Alchemy [63] is a web3 development tool, that deploys the blockchain smart contract.

To use Alchemy, it is necessary to create an application on their website. Here, the network where the app is deployed to is decided. There is the Mainnet network, the primary public Ethereum blockchain, where there are real cryptocurrency transactions (ETH), but there are also several Ethereum testnets, networks used by developers to test their dApps before deploying them to the Mainnet. As mentioned before, the application of this thesis will be tested in a Testnet network.

The foundation of the UI is presented in Figure 4.2. At the top, there is a small description of the SC deployed. The hourly electricity prices from the SPOT market, as well as the average price for the day in question, are always shown. This will help users to see the price that the SC is using at any time of that day, making it easier to define intervals where selling energy or buying energy is more advantageous. The database is displayed to help with the testing phase of this project. Finally, at the bottom, the blockchain created with the SC is visible.

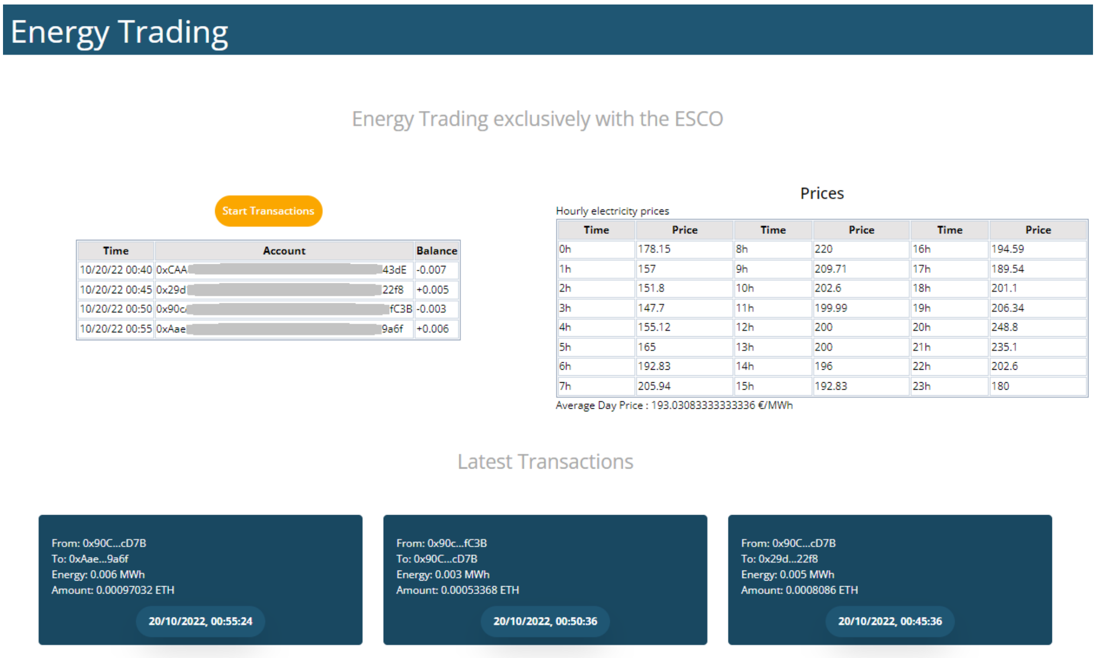

Figure 4.2: Design of the generic UI.

4.4 User's Wallet

Users of this application will need to interact with the website and actually be able to transfer funds between them, therefore a cryptocurrency wallet is necessary. Metamask is then selected to handle the cryptocurrency.

Metamask [64] allows users to interact with the Ethereum blockchain dApp. It can be accessed through a browser extension or a mobile app. Its users can securely connect to decentralized applications, and send and receive Ethereum-based cryptocurrencies and tokens.

All participants in the community, including the ESCO, can create an account with Metamask and fund their wallets with ETH.

The website where the application is running can connect to the users' accounts and integrate the smart contract functionality with the Metamask wallet, via JavaScript code that uses this wallet as an intermediary to send transaction requests to the users.

On the client side, this extension also provides a secure interface, since the user has control over their own account. The user chooses if they wish to connect to a website, and once the smart contract requests a transaction from the user through Metamask, it is up to the user to approve it or reject it. A confirmation from the sender's accounts is always necessary to proceed with a transaction.

While testing the application with Metamask, a challenge arose. It is only possible to have one active account at a time, one account that is connected to the website and the smart contract can access it to make the transaction's request to the sender. Given that authorization from the sender is required at all times, this poses a problem when there is more than one transaction to be made at the same time. Note that this is not a real-world problem, given that a participant in the community will only have one account. However, for the sake of testing this platform, where all accounts will be managed by the same person, this poses a problem. A simple solution is to make sure that there are no simultaneous transactions. Even if it is suppose to happen at the same time, it is given an interval between transactions, to give enough time to change from one account to the other.

As stated before, the application will be tested in a Testnet network. These networks allow developers to test their dApps before deploying them into the Ethereum mainnet, without having to worry about the costs of transactions while testing.

The Goerli Testnet Network was chosen amongst other testnets due to its similarity to the Ethereum mainnet. Developers can build their dApps in the Goerli network, a ledger that is separate from the main Ethereum Ledger. It runs on a proof-of-stake consensus algorithm. The validators on the Goerli testnet stake capital, in the form of Goerli ETH (from here on out, mentioned only as ETH), a fake testnet ETH, that will act as collateral and can be destroyed if the validator behaves maliciously.

The developers can request Goerli ETH to an account of their choice using a tesnet Goerli faucet, available on Alchemy.

For each of the users described in section 3.1, an account was created. These accounts are presented in the table below.

User	Account
A	0xCAA...43dE
B	0x29d...22f8
C	0x90c...fC3B
D	0xAae...a6f
ESCO	0x90C...cD7B

Table 4.1: Accounts' addresses created with Metamask.

There is also a platform available where all transactions in the Ethereum network can be seen, the Etherscan [65].

With this website, Metamask users can see all transactions made from their accounts. In Figure 4.3, an example of a transaction made in the Goerli network is shown. All details from the transaction can be seen, such as the block information, the amount of ETH transferred and the transaction fee (gas balance).

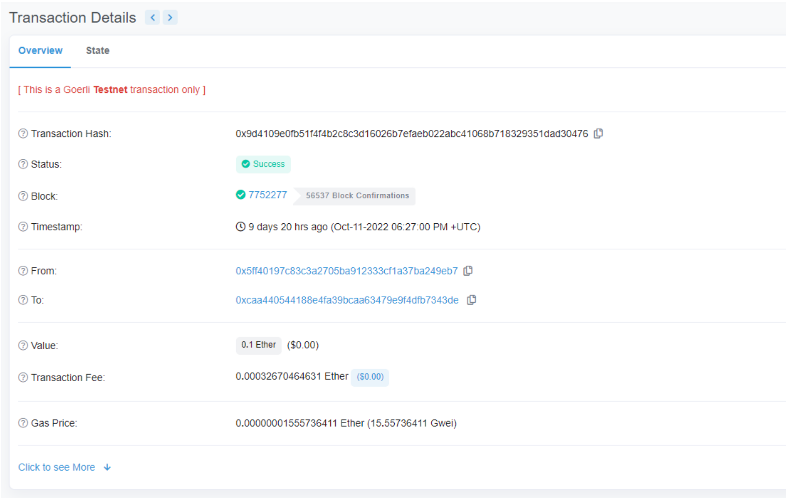

Figure 4.3: Representation of a transaction on Etherscan [65].

4.5 First Smart Contract

4.5.1 Implementation

In the first smart contract, only trading with the ESCO was allowed. The participants in the energy trading community can buy energy from the ESCO or sell their energy to the ESCO, following the contract described in section 4.1.

The application is connected to a database where all data regarding the energy trading is detailed [1]. Each transaction request has the following details: time, account, and balance. The time indicates at what time the trading is planned to happen, the user can program the energy trading to occur at a time of their preference; the account indicates the address of the user that wishes to do the trading; and the balance defines the amount of energy to trade, given the '+' signal for selling to the ESCO and the '-' for buying from the ESCO. Before starting the transaction, all the information in the database needs to be checked, to make sure that the users are prepared to receive or send the amount of energy requested.

When the time in the database corresponds to the current time, the program will execute the smart contract for that user. After it calculates the amount of ETH that the sender needs to pay for the requested amount of energy at that time, it starts the process of the transaction. First, it will ask permission from the sender, then the blockchain creates the transaction. After the validation of the transaction is completed, a new block is added to the blockchain. Then the infrastructure can proceed to transfer the amount of energy agreed upon.

In Figure 4.4, there is a representation of the separate steps of this smart contract in sequential order.

The UI used for this model is presented in Figure 4.5. The button 'Start Transactions', the only interaction the users have with the program, signals to the program that it can read the database and start the transactions.

The database (Table 4.2) that is connected to the program contains the balances already discussed in

[1] While testing, a simple CSV file is used to simulate the database, instead of the user inputting the data directly in the app or an automatic input from the home management system.

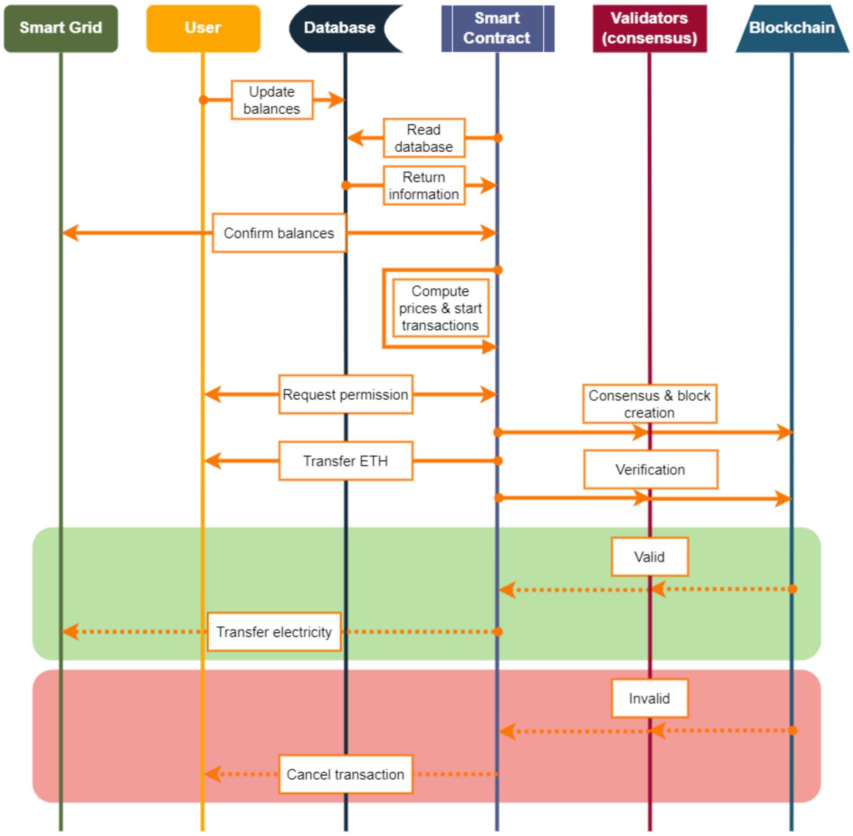

Figure 4.4: Flowchart of the first smart contract.

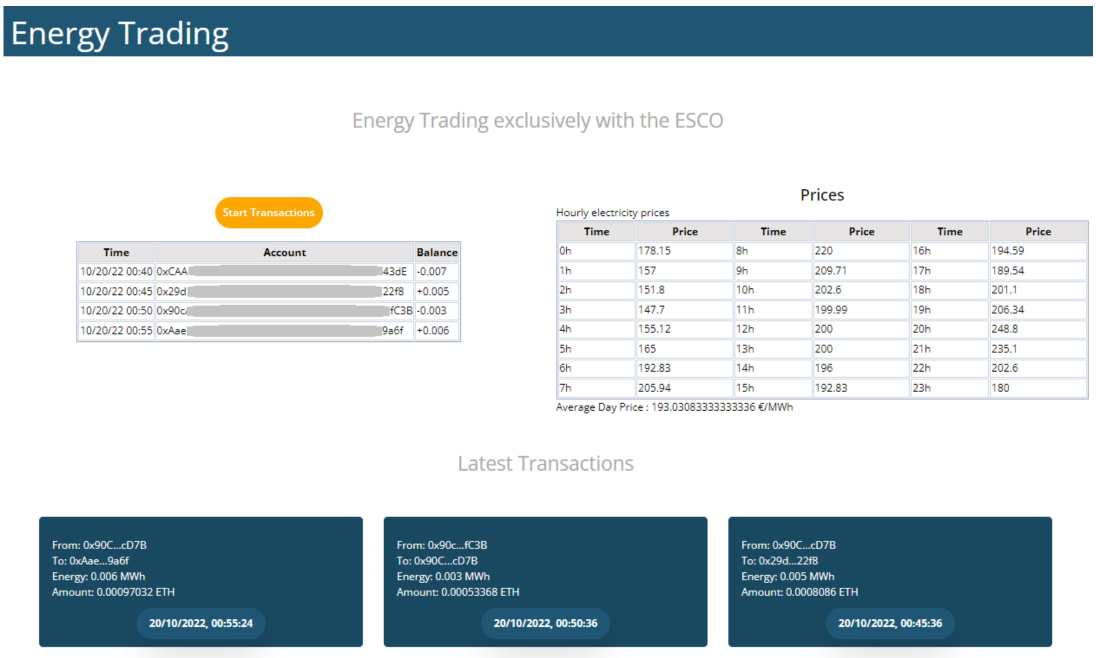

Figure 4.5: Design of the UI used for the first smart contract.

section 3.1. The time is dependent on the moment the test is being run [2]. To maintain the trial as real as possible, when applying the electricity prices during the day, the test was conducted during the day,

[2]Each transaction was made within an interval of 5 min from each other, to give time to approve the transaction and switch accounts.

and for the electricity prices at night, the test was conducted at night. However, it was not done at the exact hour of the day from when the value was extracted.

Time	Account	Balance (MWh)
t	0xCAA...43dE	- 0.007
t+5	0x29d...22f8	+ 0.005
t+10	0x90c...fC3B	- 0.003
t+15	0xAae...9a6f	+ 0.006

Table 4.2: Database loaded in the application for the first smart contract.

4.5.2 Results

The following blockchain (Figure 4.6) was created for all transactions made in the final testing phase, including all four electricity price scenarios.

Figure 4.6: Blockchain with the results of the first smart contract, when testing all electricity price scenarios.

All transactions began at the time inserted in the database, and to the correct accounts. The timestamp in the blocks indicates that all transactions were completed within one minute, proving that the gas offered, 21 000 Gwei, was enough to complete the transaction at a reasonable time.

With every transaction, there are two fees to be paid: the gas fee for the energy trading, and the contract interaction fee. Table 4.3 describes all values paid for the energy trading. The value represents the amount of ETH the buyer sent to the seller for the requested amount of energy. The TXN fee is the gas used to validate that transaction (paid to the validators). The contract fee is the fee paid to interact with the smart contract. And finally, the price paid for the energy requested (the value column in €) and the total cost of the transaction (including all fees) are also given.

From	To	Value (ETH)	TXN fee (ETH)	Contract (ETH)	Price (€)	Total cost (€)
A	ESCO	0.00050129	0.00163032	0.01292187	0.7712	23.1592
ESCO	B	0.00032552	0.00163477	0.01156920	0.5008	20.8146
C	ESCO	0.00021484	0.00157546	0.01114769	0.3305	19.9046
ESCO	D	0.00039062	0.00156820	0.01109809	0.6010	20.0876
A	ESCO	0.00110756	0.00152726	0.01080748	1.7039	20.6805
ESCO	B	0.00071919	0.00162819	0.01122015	1.1065	20.8731
C	ESCO	0.00047467	0.00151432	0.01021791	0.7303	18.7798
ESCO	D	0.00086303	0.00152931	0.01082199	1.3277	20.3297
A	ESCO	0.00059943	0.00003392	0.00023995	0.9222	1.3435
ESCO	B	0.00038924	0.00003379	0.00023745	0.5989	1.0161
C	ESCO	0.00025690	0.00003315	0.00023456	0.3952	0.8071
ESCO	D	0.00046709	0.00003370	0.00023901	0.7186	1.1382
A	ESCO	0.00124524	0.00003304	0.00023441	1.9158	2.3272
ESCO	B	0.00080860	0.00003266	0.00023110	1.2440	1.6498
C	ESCO	0.00053368	0.00003223	0.00022880	0.8210	1.2226
ESCO	D	0.00097032	0.00003209	0.00022749	1.4928	1.8922

Table 4.3: Amount of currency paid for each transaction made, in ascending time order, while testing the first smart contract.

The prices paid for the trade are according to the generic smart contract with the ESCO, explained in section 4.1. At low prices, the smart contract used the daily average to calculate the price, whereas the electricity price for the moment was used for the high electricity prices.

Note that the total cost during the day is much higher than during the night. This proves that the hour of the day when the transaction happens has a major influence on the fees applied.

4.6 Second Smart Contract

4.6.1 Implementation

In the next step, it was added peer-to-peer trading with specific rules. The ETH to be transferred is calculated by the same rules as the generic smart contract with the ESCO. The website, presented in Figure 4.7, is identical to the last.

This smart contract will trade energy in intervals. Users can request energy or the HMS can automatically input the request. The database will have the same structure as before [3]. At the end of each time interval, the program will take all the energy requests inside that interval and create a list of transfers between users.

The list of transfers will be created using the following rules:

- P2P trading has priority.
- Whoever is first in the database, i.e., whoever requested to buy or sell energy first has priority over their peers.
- At last, trading with the ESCO will be used when there is no more possible trading between peers.

[3] A CSV file with time, account and balance.

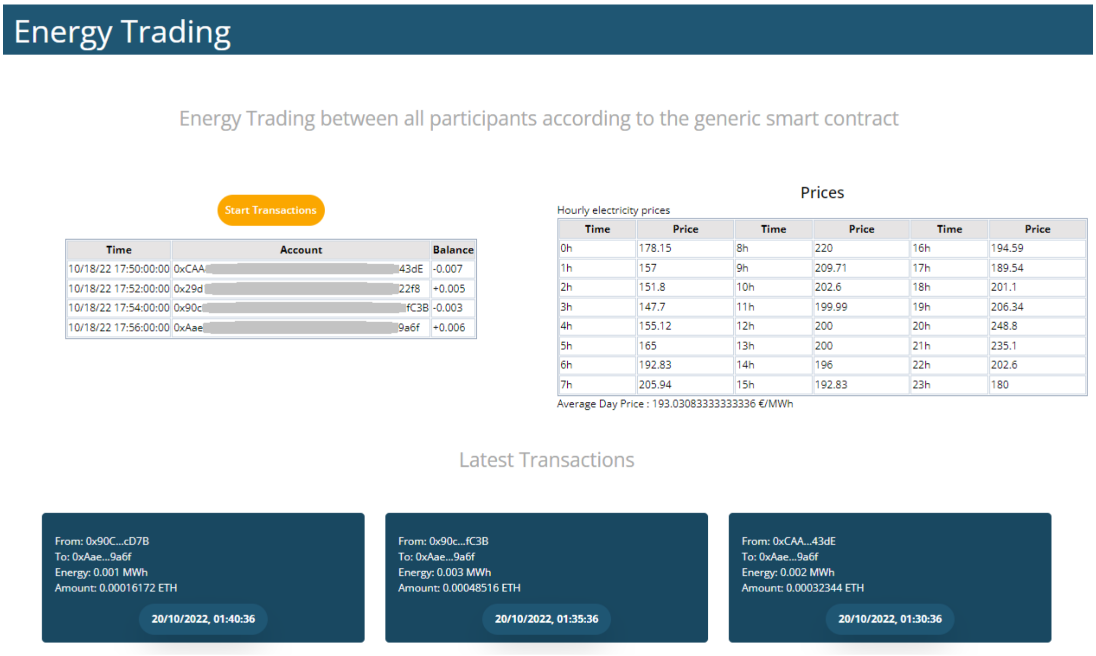

Figure 4.7: Design of the UI used for the second smart contract.

Following the rules above, the program will go through the list of the balances until it finds the first user that asked for energy, that is, the balance is negative. Now it needs to find users that can sell energy. Again, it will go through the list of balances until it finds a user that wants to sell (positive balance), and then it will create a transfer between the selected users, and update their balances. Given that whoever comes first in the file has priority, the first user that wants to sell energy will sell to the first user that wants to buy energy. If the amount of energy that the seller has is not enough, then the program will move on in the list until it finds another positive balance and creates another transfer with the amount of energy that is lacking. After the first buyer that appears on the list has the total amount of energy that he requested, the program will move on to the next buyer and repeat the process, until it reaches the end of the list.

If, at the end of this loop, there are users that still do not have the balance at zero, that is, there are users that still need to buy or sell energy, then the program will create a transfer between those users and the ESCO directly.

With the list of transfers created, which now has the following data: sender, receiver and amount of energy to transfer, the program will run the generic smart contract to calculate the prices and complete the transaction [4].

The flowchart for this smart contract is presented in Figure 4.8.

Using the same database inserted in the first smart contract, replicated again in Table 4.4, these are the steps that the program follows to create the list of transfers:

1. Reading the balance column of the database, it finds the first value of energy that indicates that the user wishes to buy energy (negative value). In this case, is the first row, with user A wanting

[4] Once again, a timer of 5 minutes is set between transfers, to give enough time to switch accounts to give permission to proceed with the transaction.

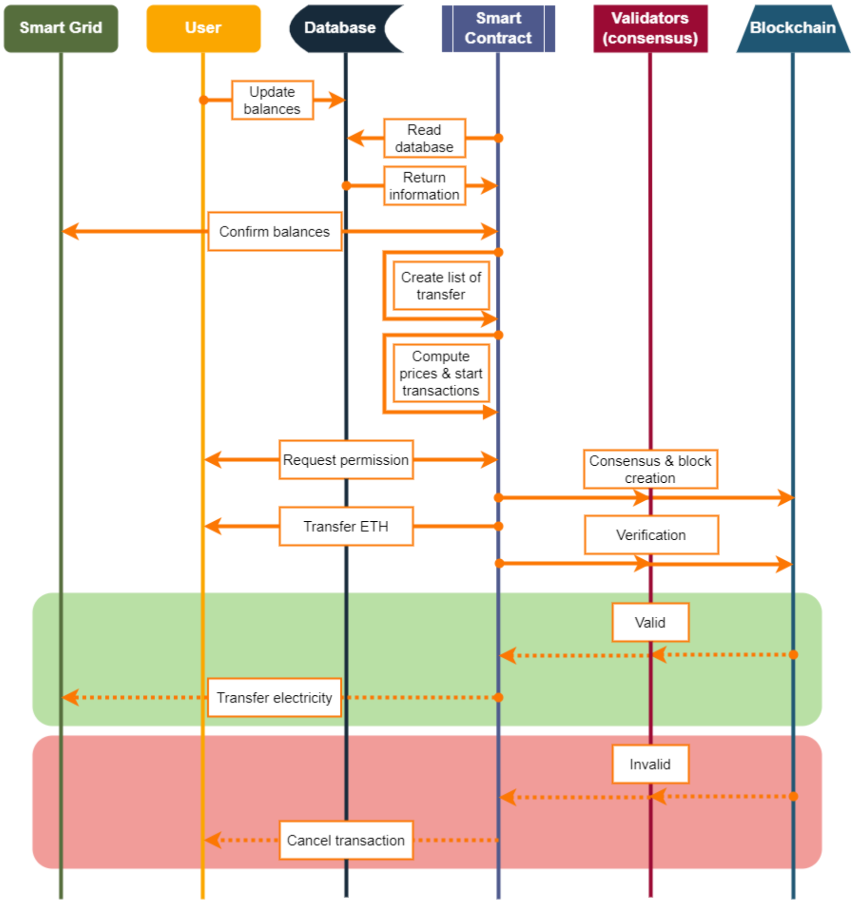

Figure 4.8: Flowchart of the second smart contract.

Time	User	Balance (MWh)
t	A	- 0.007
t+5	B	+ 0.005
t+10	C	- 0.003
t+15	D	+ 0.006

Table 4.4: Database loaded in the application for the second smart contract.

to buy 0.007 MWh.

2. Then, goes back to the beginning to find the first positive value, which is user B that wishes to sell 0.005 MWh. It creates a transfer between user A to user B, for 0.005 MWh of electricity, and updates new balance values: user A still needs 0.002 MWh and user B has nothing left to sell.

3. Since user A still needs energy, it goes through the list one more time to find another user that wants to sell energy, finding user D with 0.006 MWh to sell. It creates another transfer, between user A to user D, for 0.002 MWh of electricity. The new balances for users A and D are, respectively, 0 and +0.004 MWh.

4. With user A satisfied, it goes back to find if there is another user that needs electricity. It finds user C, that has requested 0.003 MWh. With only user D left to sell electricity, it adds another transfer to the list, between user C to user D, for 0.003 MWh, and updates new balances.

5. Considering that there are no more users that need to buy energy, it goes through a final sweep of the database, to see if any users have a balance different from zero. It finds user D still with 0.001 MWh to sell, which will be sold to the ESCO since that are no more peers that want energy. A final transfer is created, between ESCO to user D, to buy the last 0.001 MWh in the community.

This leads to the following list of transfers:

Account from	Account to	Balance (MWh)
A	B	0.005
A	D	0.002
C	D	0.003
ESCO	D	0.001

Table 4.5: List of transfers created by the program when executing the second smart contract, for the database presented in table 4.4.

4.6.2 Results

The blockchain presented in Figure 4.9 was created for all transactions made in the testing phase, for the second smart contract and for all electricity prices scenarios.

Figure 4.9: Blockchain with the results of the second smart contract, when testing all electricity price scenarios.

The smart contract was able to create the correct list of transfers to execute (confirm in Table 4.5).

Once again, several fees were paid to make these transactions associated with the Ethereum Blockchain. Table 4.6 describes the results for the second smart contract.

Note that this contract, while satisfying the existence of P2P trading, it is not the best logic to trade amongst peers. If a buyer is asking for a larger quantity than any offer available, which is the case of user A, he will have to trade twice as many (or more times) to get the energy he needs. Consequently,

From	To	Trading (ETH)	TXN fee (ETH)	Contract (ETH)	Price (€)	Total cost (€)
A	B	0.00032552	0.00154456	0.01396835	0.5008	24.3668
A	D	0.00013021	0.00157685	0.01115746	0.2003	19.7916
C	D	0.00019531	0.00151529	0.01072187	0.3005	19.1269
ESCO	D	0.00006510	0.00161863	0.01056643	0.1002	18.8464
A	B	0.00071919	0.00141075	0.00998298	1.1065	18.6353
A	D	0.00028768	0.00106435	0.00753172	0.4426	13.6673
C	D	0.00043152	0.00063583	0.00460991	0.6639	8.7342
ESCO	D	0.00014384	0.00045356	0.00305601	0.2213	5.6206
A	B	0.00003892	0.00003170	0.00022414	0.5989	0.4535
A	D	0.00015570	0.00003164	0.00022385	0.2395	0.6326
C	D	0.00023354	0.00003155	0.00022323	0.3593	0.7513
ESCO	D	0.00007785	0.00003153	0.00022309	0.1198	0.5115
A	B	0.00080860	0.00003151	0.00022298	1.2440	1.6355
A	D	0.00032344	0.00003151	0.00022293	0.4976	0.8890
C	D	0.00048516	0.00003151	0.00022292	0.7464	1.1378
ESCO	D	0.00016172	0.00003151	0.00022289	0.2488	0.6402

Table 4.6: Amount of currency paid for each transaction made, in ascending time order, while testing the second smart contract.

user A paid twice as many fees as the other peers. Analyzing the results, it would be more beneficial to trade the full amount directly with the ESCO, instead of paying twice as many fees.

4.7 Third Smart Contract

4.7.1 Implementation

For the last smart contract, the P2P trading is done through a first-price sealed-bid auction. The design for this website is presented in Figure 4.10.

In this smart contract, there are three main features:

- Auctions

The user can decide to start an auction, by inputting 3 values: the amount of energy he wants to sell in that auction, the amount of time he wishes for the auction to be open, and the minimum bid, the minimum amount of ETH the bidders can bid. Once the time of the auction has ended, the winner will be selected, and the transaction will be made [5]. The ESCO cannot participate in the auctions.

Users can see the auctions that are available. Once an auction has started, the owner can also choose to cancel the auction. Only one auction per user at one time is allowed. And to avoid high values of minimum bids, if an auction has come to an end and there are no bids, the energy will be sold directly to the ESCO, at the price in the generic contract.

- Bids

The users can place bids anonymously in the open auctions, simply by giving the address of the auction's owner and their bid (in ETH). They can place as many bids as they wish on the available

[5]Ideally, the time feature is working. However, while testing, this feature proved to be inefficient. So, the option to end the auction automatically is added, to overpower the time of the auction.

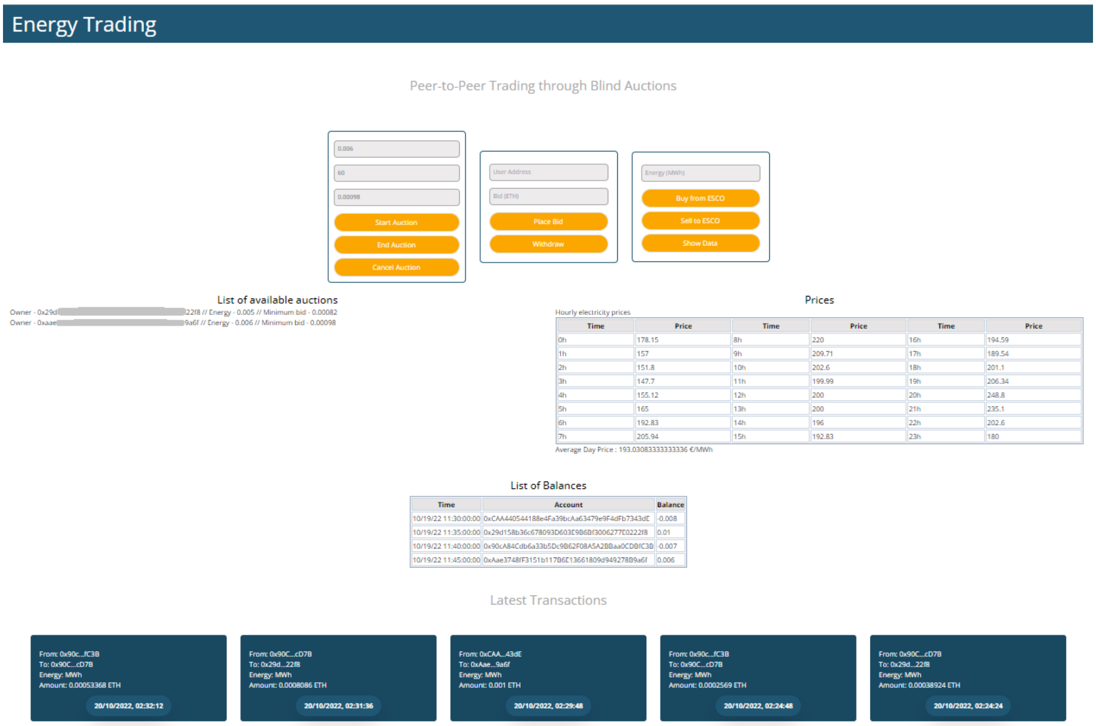

Figure 4.10: Design of the UI used for the third smart contract.

auctions. If an auction ends and the highest bid belongs to more than one user, then the winner will be the user that placed the bid first.

There is also the option to withdraw from an auction. When an user gives the auction's owner address, all bids made in that auction will be deleted.

- Trading with ESCO

At last, the users can also trade directly with the ESCO. If the user does not want to participate in auctions or the trade is time sensitive, they can trade directly with the ESCO, at the agreed price in the generic contract.

It is assumed that buying from the ESCO has an energy trading fee of 10%, this leaves the users with enough room to make their trading amongst peers without losing money. If a user wants to buy energy from the ESCO, it will pay the price at the moment (or the daily average) plus a 10% fee, having to pay 1.1 of the electricity price. If a user wishes to sell energy to the ESCO, it will only receive the electricity price. This means, that if the user wants to receive a bigger compensation for their electricity, it has that 10% of freedom to trade amongst peers, without any loss to either participant.

In Figure 4.11, there is a representation of sequential steps for this smart contract. The flowchart only represents the steps for the auctions. When trading directly with the ESCO, the program will follow the steps represented in the first smart contract (Figure 4.4), but instead of reading the balance of a database, the user can request it directly on the website.

As it was described, this smart contract presents an increased complexity. There are more possible options for energy trading. As a result, two different situations are considered, in order to test the most features of the application: 2 auctions available and both have bids, and 2 auctions available but only

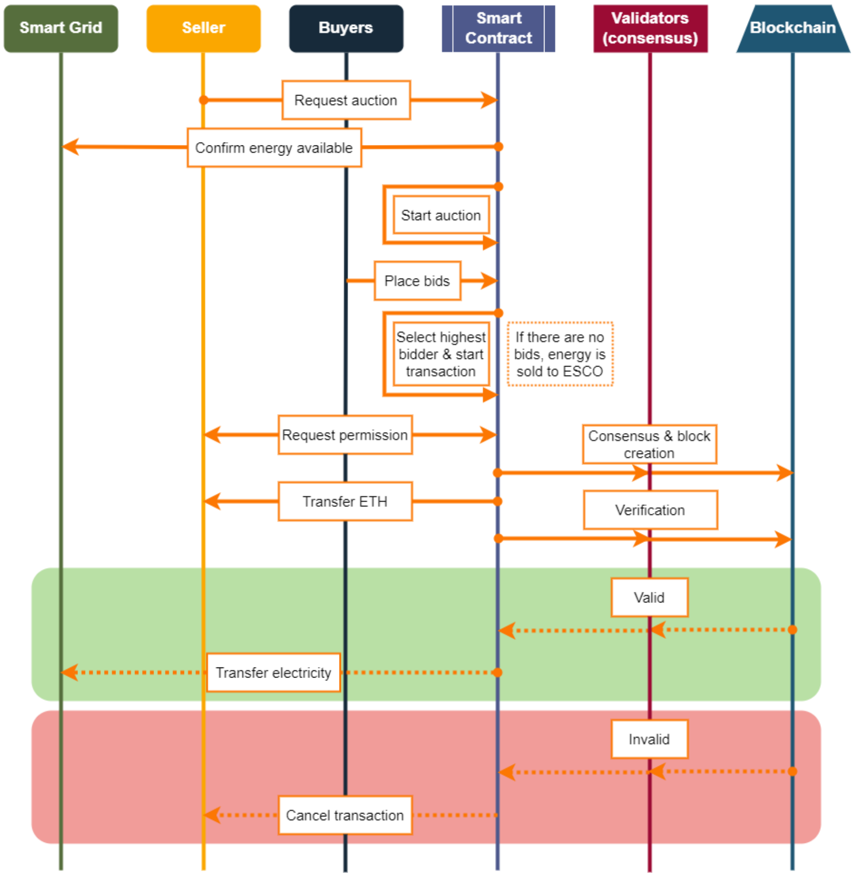

Figure 4.11: Flowchart of the third smart contract.

one has bids.

Situation 1

User B and user D have auctions available at reasonable prices, i.e., between 1 and 1.1 of the electricity price at that moment. Therefore, user A and user C place bids in both auctions, user A places the highest bid in the user D auction, and user C places the highest bid in the user B auction.

In Table 4.7, the values for the auctions and bids used while testing are presented.

Situation 2

User B and user D have auctions, but user B is asking more than the ESCO at that time. Therefore, user A and user C only place bids in user D auction, where user A places the highest bid. Since the auction of user B does not have any bids, the energy will be sold directly to the ESCO. User C buys energy directly from the ESCO.

In Table 4.8, the values for the auctions and bids are presented.

Account	Electricity (MWh)	Minimum bid (ETH)
Day – low prices		
B	0.005	0.000333
D	0.006	0.000397
Day – high prices		
B	0.005	0.000730
D	0.006	0.000880
Night – low prices		
B	0.005	0.000395
D	0.006	0.000480
Night – high prices		
B	0.005	0.000820
D	0.006	0.000980

((a)) Auctions.

Account	Auction B	Auction D
Day – low prices		
A	0.000339	0.000418
C	0.000340	0.000400
Day – high prices		
A	0.000740	0.000917
C	0.000755	0.000903
Night – low prices		
A	0.000405	0.000504
C	0.000409	0.000492
Night – high prices		
A	0.000843	0.001000
C	0.000860	0.000993

((b)) Bids, in ETH.

Table 4.7: Values of auctions and bids used while testing the third smart contract, for the first situation.

Account	Electricity (MWh)	Minimum bid (ETH)
Day – low prices		
B	0.005	0.000360
D	0.006	0.000397
Day – high prices		
B	0.005	0.000800
D	0.006	0.000880
Night – low prices		
B	0.005	0.000432
D	0.006	0.000480
Night – high prices		
B	0.005	0.000900
D	0.006	0.000980

((a)) Auctions.

Account	Auction B	Auction D
Day – low prices		
A	-	0.000418
C	-	0.000400
Day – high prices		
A	-	0.000917
C	-	0.000903
Night – low prices		
A	-	0.000504
C	-	0.000492
Night – high prices		
A	-	0.001000
C	-	0.000993

((b)) Bids, in ETH.

Table 4.8: Values of auctions and bids used while testing the third smart contract, for the second situation.

4.7.2 Results

When the third and last smart contract was implemented, the blockchain in Figure 4.12 was created with all transactions made while testing situation 1 of the third SC, for all electricity price scenarios.

In this situation, using the data from Table 4.7, user A won the auction of user D, and user C won the auction of user B, as predicted. Note here, that the balances of the users were taken into consideration as best as possible, but this leaves with the updated balances different than in the previous two smart contracts. For example, user B is selling 0.005 MWh, and user C wins that auction, even if he only wanted 0.003 MWh.

Table 4.9 describes the results for the third smart contract, when testing the first situation.

Figure 4.12: Blockchain with the results of the third smart contract, when testing all electricity price scenarios and the first situation.

From	To	Trading (ETH)	TXN fee (ETH)	Contract (ETH)	Price (€)	Total cost (€)
A	D	0.000418	0.00057388	0.00557824	0.6431	10.1079
C	B	0.000340	0.00059526	0.00421232	0.5231	7.9194
A	D	0.000917	0.00065155	0.00461059	1.4108	9.5064
C	B	0.000755	0.00062418	0.00441696	1.1615	8.9171
A	D	0.000504	0.00003150	0.00022291	0.7754	1.1668
C	B	0.000409	0.00003150	0.00022291	0.6292	1.0206
A	D	0.001000	0.00003150	0.00022291	1.5385	1.9299
C	B	0.000860	0.00003150	0.00022291	1.3231	1.7145

Table 4.9: Amount of currency paid for each transaction made, in ascending time order, while testing situation 1 of the third smart contract.

In this situation, only two transactions per electricity scenario were made. The winner of the auctions paid the exact amount of ETH they bid. Even if user A still needed 0.001 MWh according to their balance, no adjustment was made.

Regarding situation 2, the blockchain created is presented in Figure 4.13. And in Table 4.10, the results are laid out.

Figure 4.13: Blockchain with the results of the third smart contract, when testing all electricity price scenarios and the second situation.

From	To	Trading (ETH)	TXN fee (ETH)	Contract (ETH)	Price (€)	Total cost (€)
A	D	0.00041800	0.00027795	0.00196687	0.6431	4.0966
ESCO	B	0.00032552	0.00032440	0.00218754	0.5008	4.3653
C	ESCO	0.00021484	0.00027401	0.00215068	0.3636	4.0608
A	D	0.00091700	0.00030241	0.00184237	1.4108	4.7104
ESCO	B	0.00071919	0.00025729	0.00196836	1.1064	4.5305
C	ESCO	0.00047467	0.00025774	0.00182385	0.8033	3.9327
A	D	0.00050400	0.00003150	0.00022291	0.7754	1.1668
ESCO	B	0.00038924	0.00003150	0.00022291	0.5988	0.9902
C	ESCO	0.00025690	0.00003150	0.00022290	0.4348	0.7866
A	D	0.00100000	0.00003150	0.00022291	1.5385	1.9299
ESCO	B	0.00080860	0.00003150	0.00022290	1.2440	1.6354
C	ESCO	0.00053368	0.00003150	0.00022291	0.9032	1.2124

Table 4.10: Amount of currency paid for each transaction made, in ascending time order, while testing the second situation of the third smart contract.

For each electricity price scenario, three transactions are made, one for the auction of user D, where user A won and paid the exact amount of ETH he bid. Since user B auction did not receive any bids, the energy was sold directly to the ESCO, and user C bought energy from the ESCO. When trading with the ESCO, the generic smart contract computed the prices.

Chapter 5

Discussion

5.1 Results

All smart contracts developed for this dissertation proved to be effective when testing them. All transactions had the correct sender and receiver, and the smart contract computed the correct price in all electricity price scenarios.

Comparing the three smart contracts deployed, it is obvious that the third one, where the trade is done through a blind auction, is the most favorable one. In the last smart contract, a seller can sell their energy at a higher price than the electricity price at that moment, which is all that the ESCO is willing to pay. And since the ESCO asks for 1.1 of the electricity price, it is more profitable for buyers to participate in the auctions. If a seller wants to ask more than the ESCO, it can do so, but with the risk of the energy being sold directly to the ESCO for the agreed price, at the end of the auction. This pushes the seller to sell their energy with a minimum bid between 1 and 1.1 of the electricity price, and buyers to bid in the auctions available, instead of automatically paying 1.1 of the electricity price to the ESCO.

Analyzing the tables with the results for all tests that were performed, the first aspect which is important to notice is the fees that the user must pay to operate with this application. This application uses the Ethereum blockchain, and every time the user is making a transaction using a smart contract, two fees must be paid:

- Transaction fee associated with the energy trading: the gas paid to the validators for the transaction.
- Contract interaction: the fee paid to interact with the contract.

The transaction fees are linked to the congestion in the network. The more people transact, the more congested the network becomes, following higher gas fees to incentivize the validators to work faster. On the other hand, the contract interaction is directly connected to the complexity of the smart contract, as well as memory usage and other factors.

In this particular case, these compensations paid to the network in a single transaction are already higher than the amount of ETH paid for the energy trading, the user is paying more to use the application than for the energy he requests in each transaction.

In Table 5.1, the maximum and minimum amount paid in fees in the transactions are presented, to evaluate the range of these fees, distinguishing when the transaction occurs during the day, or at night.

Day	Maximum (ETH)	Minimum (ETH)
Transaction fee	0.00163448	0.00025730
Contract interaction fee	0.01396835	0.00182385
Night	Maximum (ETH)	Minimum (ETH)
Transaction fee	0.00003391	0.00003150
Contract interaction fee	0.00023995	0.00022289

Table 5.1: Maximum and minimum fees, in ETH, paid for the transactions, differentiating transactions during the day and during the night.

During the night, the fees paid were considerably lower than during the day. Since the highest amount of ETH transacted between participants was 0.00124524 ETH, for 0.007 MWh of electricity, this means that the moments where it is beneficial to trade in this application are constricted. It is only favorable to trade energy in this application when the gas fees are low (mainly at night) or when trading larger amounts of energy, so that the fees are lesser than the ETH paid for the energy trading.

Overall, this application seems to have two critical elements that determine if the moment is convenient for energy trading: the electricity prices, where one needs to see when the prices are low (including the daily average) and gas prices, to make sure that the fees paid for the transaction do not overpower the actual price transacted.

5.2 Real World Application

Since this dissertation focuses primarily on designing smart contracts, that use blockchain technology, for energy trading, in this subsection, the implementation of this application in the real world is quickly analyzed.

First, the individuals that can participate in this network must be defined. This application is prepared to receive all types of participants: consumers, prosumers and even producers. However, a local energy community needs to be outlined, considering that there would be no point in P2P trading where the house of the peers would be far from each other, since there would be additional costs and losses in the transportation of the electricity. So, the participants that wish to join the community need to be reasonably close to their peers.

The application has to be restricted to the participants in the energy community. Unauthorized individuals should not get access to the application.

An Energy Service Company, ESCO, needs to be associated with the project from the beginning. It would supply all the equipment and infrastructures necessary for this energy community. The ESCO would only provide a single smart contract for all participants in the community. Everyone would have to agree to the terms bestowed by the company.

As new participants join the energy trading community, they would have to provide some personal information, such as identification, address of the household, and their cryptocurrency details. With private information in the network, security needs to be a focal point when designing the application. This information should be private, not public to all participants, and can even be stored outside the blockchain.

The ESCO is assigned the responsibility to verify if all users are ready to participate in energy trading. All participants have to be connected through a smart grid infrastructure. All houses and the ESCO are connected by cables that transport electricity from one location to the other. Each house needs to have an electricity storage that is also connected to the common grid. At the entrance and exit of this structure, a smart meter is required to be installed to connect with the application. This device will measure the electricity flow, making sure that the electricity that is supposed to be traded, is effectively relocated.

By inserting the community in a smart grid infrastructure, it is also possible to improve the energy efficiency in all houses. Aside from the required smart meter, participants can also install other smart meters around the house, and a Home Management System. This system can then take the smart meter data and make recommendations to improve energy efficiency.

With either of the smart contract designs, the participants could trade energy according to their own preferences. They can choose the time when it is better to sell or buy energy, and choose the amount of energy to trade.

As mentioned in the previous section, the fees applied while using the Ethereum blockchain appear to be a complication in making this solution alluring when trading electricity in a local community. These trading systems are normally fitted for continuous trading, meaning that the energy trading is probably not in large quantities, and users should not be obligated to have a massive storage in their homes.

These fees proved to be substantially high while trading small amounts of electricity. The maximum gas fee was settled at 21 000 Gwei, yet a complicated transaction can cost way more to validate within a reasonable time. In this project, simple transactions are made, setting the gas price at the lower end of the spectrum. Which consequently means that the fees that can be paid to trade small amounts of electricity can be much higher than the transaction. Another important aspect is to keep the smart contract as simple as possible, to assure low contract interaction fees, while still making sure that it remains secure for energy trading.

When using this application, the user needs to, not only consider the electricity price, but also the gas price, when trading. As shown in the results, when the night tests were conducted, the transaction fees were much lower than during the day. It is possible to track gas prices in the Ethereum network [66]. The price is always changing by the seconds, yet network congestion can also be somewhat predicted. In Figure 5.1, there is a heat-map representation of the hourly gas prices by days of the week.

The average gas price varies constantly, even from minutes apart. Working days tend to have higher gas prices, while during the weekend the price seems to be low throughout the day. Afternoons and early evenings have the highest price, and around 12am, the gas price starts to decrease. Be that as it may, statistically, it has been found that the lowest gas prices can be found around midday.

With these predictions, one can establish optimal times to trade, when the gas price and the electricity price are both low.

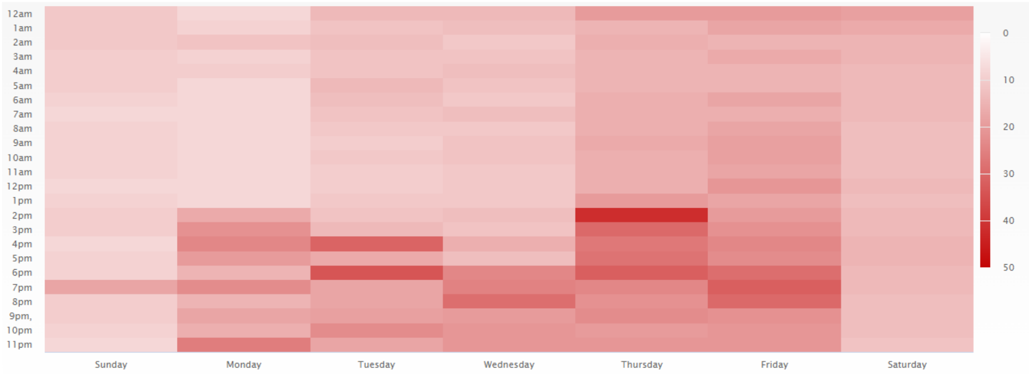

Figure 5.1: Average gas price by time of day [67].

Another influence on price is the Ethereum market. In the smart contract, the value of the € in ETH was settled as a fixed price. However, that is not the case in reality, the ETH value is always changing, as it is shown in Figure 5.2.

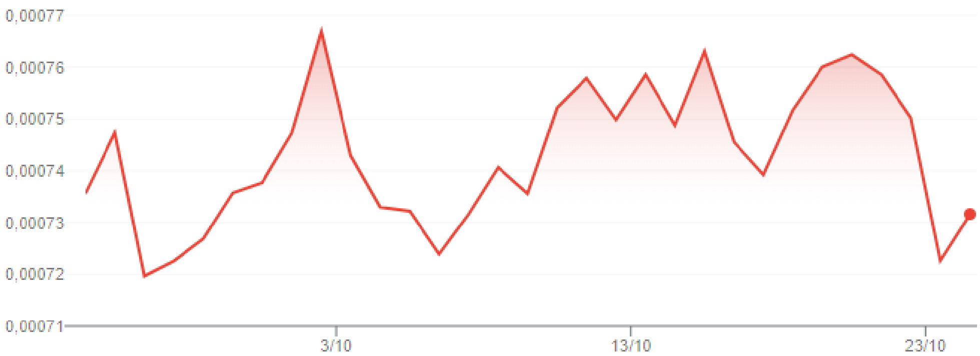

Figure 5.2: Example of the fluctuations in the exchange rate of € to ETH [68].

If a smart contract is defined with a fixed price for ETH, it may not be considered a valid contract in a month or less, when the exchange rate changes drastically.

Overall, this project shows that the Ethereum blockchain is not the most appropriate blockchain technology to use in energy trading communities. It is fundamental that the blockchain operates with fees much lower than the prices exercised in the smart contract, for example, the fees should be around 20% of the amount spent on electricity. However, this type of fees would not overpower the energy traded, if it were dealing with industrial-size amounts of energy. In that case, these fees would not have such a great negative impact.

With this specific blockchain, there is also an added issue, privacy and security, since it is a public blockchain. All the information that is stored in the blockchain is visible to all inside the network, and cannot be removed once it is a part of the network, which can create problems with privacy.

Nevertheless, blockchain-based smart contracts are still a viable option for energy trading, for maintaining a decentralized and reliable record of transactions. The blockchain guarantees the security of a record of information and it generates trust within the community, without the need for an intermediary.

Chapter 6

Concluding Remarks

6.1 Achievements

In the beginning of this thesis, it was proposed a new framework for energy trading systems: a solution to make transactions inside a local energy community. For that, a blockchain-based smart contract application was suggested. The main focus of this project was to develop smart contracts, using blockchain technology, that can be applied to P2P trading.

As the energy system transitions to more sustainable sources, it is pivotal to develop new energy frameworks, ones that can integrate new players that are becoming an important part of this ecosystem - the prosumers. These individuals need a platform were they can trade their energy with other peers, without the need for an intermediator. With new platforms, renewable energy sources can become more appealing to the everyday person, consequently increasing the consumption and production of clean energy, while weakening the dependency on fossil fuel energy. Since the residential sector accounts for the second highest consumption of electricity, this shift would have a massive impact on energy-related carbon emissions and would help to achieve the goals set out by the 2030 Agenda for Sustainable Development.

First, a generic smart contract was developed for all interactions with the ESCO, using the electricity prices from the SPOT market: if the electricity price for that moment is below 150 €/MWh, then the price would be the daily average electricity price. If the price is higher than 150€/MWh, the current price is used for the computation. If a user wishes to buy energy from the ESCO, an extra fee of 10% is applied.

After, three smart contracts were designed in Chapter 3, each one more complex than the other.

1. Trading exclusively with the ESCO - users insert their energy needs in a database, that is connected to the application. The program would then calculate the price to pay for the amount of energy requested and proceed with the transactions.

2. P2P trading according to generic contract - once again users insert their energy needs in a database, by time intervals. At the end of each interval, the program creates a list of transfers by giving priority to P2P trading and whoever requested energy first, and it would only go to the ESCO when there is no more possible P2P trade. Then, it would calculate the price and go through the transactions.

3. P2P trading through a blind auction - the users can create auctions if they wish to sell energy. Whoever wants to buy, can place bids anonymously. The highest bidder at the end of the auction wins, and the winner gets the full amount of energy that was auctioned. Additionally, it is still possible to go directly to the ESCO to trade energy.

As expected, the last smart contract is the most suited for P2P trading, since it is more advantageous for both sides. Sellers can receive a bigger payment for their energy than they would receive if they went directly to the ESCO. And buyers can pay less for the same amount of energy in auctions.

These smart contracts were run through the Ethereum Blockchain, a public blockchain with ETH (ether) as its cryptocurrency. Users can manage their ETH with the help of Metamask.

All three smart contracts created were tested accordingly to four possible scenarios for energy trading: when the electricity prices were low and high, and during the day and night. By executing the tests during the night, this also provided some insight into how the fees applied to make transactions through Ethereum behave.

By analyzing the results attained in Chapter 4, this blockchain proved to not be the most suitable platform for energy trading, where the amount of energy being traded is small, and consequently the amount of ETH to transact is extremely low. Users could be paying more for the fees applied to use the blockchain and interact with the contract, than the actual price of the energy.

Although the smart contracts were the main focus of this project, a succinct description of how to implement this application in a real-world scenario is given in the section 5.2.

Finally, it is possible to state that the objectives of this thesis were met, allowing us to better understand the energy market and the changes required to shift towards a more sustainable system. A new energy trading scheme, that is decentralized and running on secure platforms, is essential to make renewable energy, produced in the residential sector, more engaging and affordable, making blockchain-based smart contracts a promising solution for this field.

This area is still in the early stages of development and this thesis will contribute to this effort with the proposed framework to implement smart contracts within energy communities using blockchain technology.

To conclude, in the section below, some notes about the limitations, possible improvements and adjustments of this work are given, as a way to encourage the progress in this field.

6.2 Limitations and improvements

While developing the application, a few limitations came to light, which prevents this solution to be a real implementation for P2P energy trading. These constraints should be looked at as a way to improve and not as limits to its applicability.

First, the legality of a smart contract is still not well defined. Certainly, a smart contract in these conditions is not obligated to be considered as a smart legal contract. However, if a person acts against the community, consequences must be discussed. It is also necessary to have systems in place to resolve disputes. An interesting study would be to compare the energy trade using smart contracts and traditional contracts, and explore the difference in time and price.

And even though the smart contracts deployed for this dissertation were quite simple, complex smart contracts can cost a lot to deploy. Additionally, with every interaction, there is also a fee, which would also be higher. Complexity and memory allocated would have to be a relevant concern when designing the smart contract.

One possible improvement for the third smart contract would be another type of auction. In the auction deployed, the bids were placed anonymously, and the winners took all the energy that was auctioned. This is not the only type of auction that can be applied to energy trading systems. An english auction or a dutch auction can also be enforced. In an english auction, the participants bid freely and in ascending order, the last person that places a bid (the highest price) is the winner. The dutch auction operates the other way around, the auctioneer initiates the auction with a high value and the bidders will place lower bids, until a bidder is willing to end the auction. Both auctions can create more competitiveness than blind auctions. Another possible enhancement for the auction is the partition of the auctioned electricity. Imagining that a bidder only wants part of the electricity that is being auctioned, it can be possible to bid just for a part of the electricity. Then the auctioned energy would go to several bidders. This would allow the seller to auction only once, instead of auctioning small amounts of energy to guarantee that bids are placed.

Regarding the blockchain technology, Ethereum presented its own limitations. The Ethereum's cryptocurrency proved to be not the most appropriate when a small amount of ETH is being transacted, that results from small amounts of energy. The fees that are in place, such as the contract interaction, mentioned before, and the gas fee, due to network congestion, are higher than the price paid for the electricity. Consequently, making this platform not as inviting as it should be.

Nonetheless, there are strategies to help reduce gas costs. The simplest one is choosing the right moments to make a transaction. There are online tools with the gas price throughout the day, learning to predict network congestion to find the perfect moment for energy trading. It is also possible to bundle transactions together, or use other dApps, to reduce significantly the gas fees. As a final decision, one can also simulate the transaction, to see the cost of the transaction at that moment.

Another strategy would be to restructure the network, applying a two-layered solution, where the transaction information is placed off-chain and in the end, only the results are moved back into the network. Or even create a new blockchain technology, more appropriate for energy trading inside a local community.

An important concern is obviously linked to privacy and security. To participate in this energy community, its participants have to provide personal information, such as location. Since Ethereum is a public blockchain, any piece of information stored in the ledger would be visible to all, which then leads to an issue with data privacy. If a private blockchain is applied to protect sensitive information, managed for example by the ESCO, then it would not be as decentralized, since one organization would have to govern the network. It can be a mix of the two, where the personal information would be stored in a private ledger, ruled by the ESCO, while the trading would occur in a public blockchain.

Although blockchain-based smart contracts seek to pursue a decentralized solution, eliminating the need for a trusted third-party, it is almost impossible to fully eliminate them. In this case, there would be the need for an organization, ESCO, to maintain the infrastructure that develops the project.

Notwithstanding, the development of smart energy contracts, that enable users to actively participate in the energy market has to be a main concern in the present and near future. In this way, they can contribute to the increase of renewable energy production and consumption, and to a more efficient use of electricity and energy grids. And since blockchain technology was developed to implement distributed secure transactions, it is compatible with the implementation of smart energy contracts.

Clean energy technology is becoming a major new area for investment and employment – and a dynamic arena for international collaboration and competition [6]. These decentralized systems need to be designed and truly implemented, to assure consumers that this new way of handling energy is a reliable and affordable choice, hopefully turning consumers into prosumers, and consequently increase their renewable energy use and reduce their ecological footprint.

Bibliography

[1] The 17 goals — sustainable development (2022), https://sdgs.un.org/goals.

[2] U. Secretary-General, Progress towards the sustainable development goals (2021), https://digitallibrary.un.org/record/3930067?ln=enrecord-files-collapse-header.

[3] U. Nations, *The Sustainable Development Goals Report* (2021).

[4] I. A. IEA, Tracking industry 2021 (2022), https://www.iea.org/reports/industry.

[5] IEA, Global energy review 2021 (2021), https://www.iea.org/reports/global-energy-review-2021.

[6] IEA, World energy outlook 2021 (2021), https://www.iea.org/reports/world-energy-outlook-2021.

[7] H. Ritchie, M. Roser, P. Rosado, Energy (2020), https://ourworldindata.org/electricity-mix.

[8] IEA, Renewable energy market update 2021 (2021), https://www.iea.org/reports/renewable-energy-market-update-2021.

[9] IEA, Electricity – global energy review 2021 (2021), https://www.iea.org/reports/global-energy-review-2021/electricity.

[10] IEA, Key world energy statistics 2021 – analysis - iea (2021), https://www.iea.org/reports/key-world-energy-statistics-2021.

[11] IEA, Energy efficiency 2021 (2021), https://www.iea.org/reports/energy-efficiency-2021.

[12] Directive (eu) 2019/944 of the european parliament and of the council (2019), https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32019L0944.

[13] IEA, Energy efficiency 2019 (2019), https://www.iea.org/reports/energy-efficiency-2019.

[14] A. Murali, Council Post: Understanding Generation Data, *Forbes* (2021).

[15] *Global e-sustainability Initiative and Accenture, 2015*.

[16] IEA, Smart grids (2022), https://www.iea.org/reports/smart-grids.

[17] IEA, Electricity security 2021 (2021), https://www.iea.org/reports/electricity-security-2021.

[18] Electricity market design (2019), https://energy.ec.europa.eu/topics/markets-and-consumers/market-legislation/electricity-market-design.

[19] Clean energy for all europeans package (2019), https://energy.ec.europa.eu/topics/energy-strategy/clean-energy-all-europeans-package.

[20] Energy union (2015), https://energy.ec.europa.eu/topics/energy-strategy/energy-union.

[21] Protection and rights - energy consumers (2019), https://energy.ec.europa.eu/topics/markets-and-consumers/energy-consumer-rights/protecting-energy-consumers.

[22] Portugal's national energy and climate plans (2021), https://energy.ec.europa.eu/topics/energy-strategy/national-energy-and-climate-plans-necps.

[23] C. V. Miguel, A. Mendes, L. M. Madeira, An Overview of the Portuguese Energy Sector and Perspectives for Power-to-Gas Implementation, *Energies* **11** (2018).

[24] IBM, What are smart contracts on blockchain? (2022), https://www.ibm.com/se-en/topics/smart-contracts.

[25] D. Vieira Fernandes, Tokens, "Smart Contracts" and System Governance, *SSRN Electronic Journal* (2019).

[26] A. Miglani, N. Kumar, V. Chamola, S. Zeadally, Blockchain for Internet of Energy management: Review, solutions, and challenges, *Computer Communications* **151**, 395 (2020).

[27] S. Faisal, Smart contracts (2022), https://www.researchgate.net/figure/Traditional-vs-Smart-Contracts_fig21_340730760.

[28] G. Governatori, *et al.*, On legal contracts, imperative and declarative smart contracts, and blockchain systems, *Artificial Intelligence and Law* **26**, 377–409 (2018).

[29] IBM, What is blockchain technology? (2022), https://www.ibm.com/se-en/topics/what-is-blockchain.

[30] IBM, Benefits of blockchain (2022), https://www.ibm.com/topics/benefits-of-blockchain.

[31] R. Mantri, What are blockchain layers and how do they work (2022), https://coinstats.app/blog/what-are-blockchain-layers-and-how-do-they-work/.

[32] I. A. S. M. . B. R. Younes, Z., Blockchain Applications and Challenges in Smart grid (2021).

[33] L. Wang, *et al.*, A Permissioned Blockchain-Based Energy Management System for Renewable Energy Microgrids, *Sustainability* **13** (2021).

[34] Z. Zheng, S. Xie, H. Dai, X. Chen, H. Wang, An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends, *2017 IEEE International Congress on Big Data (BigData Congress)* (2017).

[35] M. Andoni, *et al.*, Blockchain technology in the energy sector: A systematic review of challenges and opportunities, *Renewable and Sustainable Energy Reviews* **100**, 143 (2019).

[36] How does a transaction get into the blockchain? (2016), https://www.euromoney.com/learning/blockchain-explained/how-transactions-get-into-the-blockchain.

[37] what is a smart contract and how does it work? (2022), https://cointelegraph.com/ethereum-for-beginners/what-are-smart-contracts-a-beginners-guide-to-automated-agreements.

[38] S. N. G. Gourisetti, *et al.*, Blockchain smart contract reference framework and program logic architecture for transactive energy systems, *Applied Energy* **304**, 117860 (2021).

[39] A. Hasankhani, S. Mehdi Hakimi, M. Bisheh-Niasar, M. Shafie-khah, H. Asadolahi, Blockchain technology in the future smart grids: A comprehensive review and frameworks, *International Journal of Electrical Power Energy Systems* **129**, 106811 (2021).

[40] F. Buccafurri, G. Lax, L. Musarella, A. Russo, An Ethereum-based solution for energy trading in smart grids, *Digital Communications and Networks* (2021).

[41] A. S. Yahaya, *et al.*, Blockchain Based Sustainable Local Energy Trading Considering Home Energy Management and Demurrage Mechanism, *Sustainability* **12**, 3385 (2020).

[42] A. Dorri, F. Luo, S. S. Kanhere, R. Jurdak, Z. Y. Dong, SPB: A Secure Private Blockchain-Based Solution for Distributed Energy Trading, *IEEE Communications Magazine* **57**, 120 (2019).

[43] G. Bansal, A. Dua, G. S. Aujla, M. Singh, N. Kumar, SmartChain: A Smart and Scalable Blockchain Consortium for Smart Grid Systems, *2019 IEEE International Conference on Communications Workshops (ICC Workshops)* pp. 1–6 (2019).

[44] J. Hwang, *et al.*, Energy Prosumer Business Model Using Blockchain System to Ensure Transparency and Safety, *Energy Procedia* **141**, 194 (2017). Power and Energy Systems Engineering.

[45] J. Mello, *et al.*, *Power-to-Peer: a blockchain P2P post-delivery bilateral local energy market* (2020), pp. 1–5.

[46] G. van Leeuwen, T. AlSkaif, M. Gibescu, W. van Sark, An integrated blockchain-based energy management platform with bilateral trading for microgrid communities, *Applied Energy* **263**, 114613 (2020).

[47] L. Wang, *et al.*, A Permissioned Blockchain-Based Energy Management System for Renewable Energy Microgrids, *Sustainability* **13** (2021).

[48] L. Wang, *et al.*, Credible Peer-to-Peer Trading with Double-Layer Energy Blockchain Network in Distributed Electricity Markets, *Electronics* **10** (2021).

[49] J. Eberhardt, M. Peise, D.-H. Kim, S. Tai, *Privacy-Preserving Netting in Local Energy Grids* (2020), pp. 1–9.

[50] M. Galici, *et al.*, Energy Blockchain for Public Energy Communities, *Applied Sciences* **11** (2021).

[51] I. Losa, *et al.*, Overview of the solutions developed by the GRID+ project for supporting the European smart grids pilot projects in achieving the EEGI goals: main findings and lessons learnt, *Comillas.edu* (2015).

[52] Power ledger (2022), https://www.powerledger.io/company/power-ledger-whitepaper.

[53] E. Mengelkamp, *et al.*, Designing microgrid energy markets: A case study: The Brooklyn Microgrid, *Applied Energy* **210**, 870 (2018).

[54] Omie spot market (2022), https://www.omie.es/pt/spot-hoy.

[55] Y.-J. Chen, J.-L. Wu, Y.-C. Hsieh, C.-W. Hsueh, An Oracle-Based On-Chain Privacy, *Computers* **9**, 69 (2020).

[56] Api - spot electricity market (2022), https://apidatos.ree.es/es/datos/mercados/precios-mercados-tiempo-real?start_date=2022-04-26T00:00end_date=2022-04-26T23:59time_trunc=hour.

[57] Ethereum, Ethereum development documentation (2022), https://ethereum.org/en/developers/docs/.

[58] web3.js (2022), https://web3js.readthedocs.io/en/v1.8.1/.

[59] React (2021), https://reactjs.org/.

[60] Vite (2019), https://vitejs.dev/.

[61] Tailwind css (2020), https://tailwindcss.com/.

[62] Hardhat (2022), https://hardhat.org/.

[63] Alchemy (2022), https://www.alchemy.com/.

[64] Metamask (2022), https://metamask.io/.

[65] Etherscan, Ethereum (eth) blockchain explorer (2022), https://etherscan.io/.

[66] Etherscan, Ethereum gas tracker — etherscan (2022), https://etherscan.io/gastrackerhistoricaldata.

[67] Ethereum gas price charts (2022), https://ethereumprice.org/gas/.

[68] Google finance - euro to eth (2013), https://www.google.com/finance/quote/EUR-ETH?sa=Xved=2ahUKEwjqs9jwq4H7AhXohP0HHYtOD-EQ-fUHegQICBAe.