



PR1-T3 Core Content

Modules 3: Smart Contracts

Author: Innovation Hive

PROJECT ID:

Grant agreement	2021-1-IE01-KA220-VET-000032943
Programme	Erasmus+
Key action	KA220-VET - Cooperation partnerships in vocational education and training
Field	Vocational Education and Training
Project acronym	TrainChain
Project title	TrainChain - Blockchain Training for Start Ups
Project starting date	28/02/2022
Project duration	24 months
Project end date	27/02/2024

Disclaimer: This project is funded with the support of the European Commission. The information and views set out in this document are those of the author(s) and do not necessarily reflect the official opinion of the European Commission. Neither the European Union institutions nor any person acting on their behalf may be held responsible for the use, which may be made of the information contained therein.

REVISION HISTORY

Version	Date	Author	Description	Action	Pages
1.0	31/07/2022	CCSDE	Creation	C	8
	31/12/2022	Innovation Hive	Insert	I	22

(*) Action: C = Creation, I = Insert, U = Update, R = Replace, D = Delete

REFERENCED DOCUMENTS

ID	Reference		Title
1	2021-1-IE01-KA220-VET-000032943		TrainChain Agreement
2			

APPLICABLE DOCUMENTS

ID	Reference		Title
1			
2			

Contents

1. Introduction	5
1.1 Module Description	5
1.2 Module Goals	5
1.3 Learning Objectives	5
1.4 Learning Outcomes	5
2. Main Content	6
2.1 State of the Art: Current situation and existing Problems	6
2.2 Blockchain and Smart Contracts	6
How can blockchain be utilized?	7
How Smart Contracts Work	7
Smart contracts in untrusted and semi trusted environments	8
2.3 Which of the problems will be addressed?	9
2.4 Real life implementations with details.	11
2.5 Proposed resources	14
3. Knowledge Assessment	17
4. Module Summary	19
5. References	20

1. Introduction

1.1 Module Description

The blockchain is a distributed database that allows anonymous and secure transactions between parties without the need for intermediaries. However, when smart contracts are applied in the context of these transactions, the use of the blockchain can be modified to offer a wider range of applications in different sectors and industries. Smart contracts are highly digital protocols that allow us to engage with high-value digital assets to make businesses faster and more agile. In the following, the uses of smart contracts when integrated into a blockchain will be analyzed, as well as the technical processes required to implement them.

1.2 Module Goals

Learn about the role of a smart contract within the blockchain and how these contracts can be used to meet business objectives. Also, understanding the technical aspect of these contracts, so that later on to be able to think critically about where they can be applied.

1.3 Learning Objectives

Providing valuable information and understanding about what smart contracts are and the logic behind them. In addition, a thorough analysis as to how such contracts can be exploited by a business in different areas and what goals they can help achieve.

1.4 Learning Outcomes

After completing this module, learners will be able to understand what smart contracts are in a blockchain and how they can be used to maximize business value. At the same time, the technical inspection of different functions and the implementation of such contracts will be analyzed in order to provide learners with in-depth knowledge. More so, real-life applications of these contracts in various scenarios are included to inspire the trainees and provide solutions.

2. Main Content

2.1 State of the Art: Current situation and existing Problems

Nowadays, transparency in data is considered a precious commodity, and especially in the new era of automation and technologies, it is a necessary asset. Thus, when wasting a lot of time and effort in duplicating records, in performing procedures that require close attention, or even when performing several tasks simultaneously, many errors and dangers may occur that compromise the validity of data. Restricted transparency can slow down the verification of data and, as result, negatively impact any business. Blockchain was created to address such problems since it can ensure data authenticity and security.

2.2 Blockchain and Smart Contracts

Many platforms and apps created using blockchain or distributed ledger technology incorporate "smart contracts." The term "smart contract" was first mentioned by a computer scientist and cryptographer Nick Szabo in 1997 long before blockchain. He called them smart because they automatically can execute certain pre-programmed steps, but they should not be seen as intelligent tools that can parse a contract's more subjective requirements. Thus, blockchain-based smart contracts are programs that launch when certain criteria are satisfied. They are often used to automate the implementation of an agreement so that all parties may be certain of the conclusion right away, without the need for an intermediary or any unnecessary delays. They can also automate a workflow, starting the subsequent step when the specified conditions are met.

How can blockchain be utilized?

For a blockchain to be more flexible and adaptable to different scenarios, smart contracts were incorporated into it. Smart contracts can be translated as efficient digital protocols that enable users to work with high-value digital assets so that businesses become faster and more agile. In particular, in a blockchain, they guarantee security since the distributed ledger is encrypted and impermeable, as well as great dependability in transactions. Furthermore, smart contracts substitute intermediary parties, saving both time and money while performing efficiently and avoiding many errors that would likely occur if everything was done manually.

How Smart Contracts Work

As mentioned before Smart Contract are self-executed statements written into code on a blockchain that are implemented when certain conditions are met. The code can either be the only embodiment of the parties' agreement or it can supplement a standard text-based contract by carrying out certain terms, such as transferring money from Party A to Party B. Because the code is copied over several nodes of a blockchain, it benefits from the security, permanence, and immutability that a blockchain provides. Most smart contracts are written in one of the programming languages directly suited for such computer programs, such as Solidity. A common example of a smart contracts code is "if { ; } else { ; }" which can be explained as "if/when...then...". For instance, a smart contract can be programmed to release funds to the appropriate parties, registering a vehicle, sending notifications, or issuing a ticket. When the transaction is completed, the blockchain is updated. This implies that the transaction cannot be modified, and the results are only visible to persons who have been granted permission.

A smart contract can have as many specifications as necessary to reassure the participants that the work will be executed correctly. Participants must identify how transactions and associated data are represented on the blockchain, agree on the "if/when...then..." rules that govern those transactions, investigate all conceivable exceptions, and design a framework for resolving disputes in order to set the terms.

Smart contracts are now best suited to autonomously executing two sorts of "transactions" seen in many contracts: (1) assuring the payment of funds upon specified triggering events and (2) imposing financial penalties if certain objective requirements are not met. In each situation, human participation, whether through a trusted escrow holder or even the legal system, is not necessary once the smart contract has been deployed and is operational, decreasing contract execution and enforcement costs.

Smart contracts in untrusted and semi trusted environments

Taking a more technical approach, smart contracts aim to ensure that transactions carried out by parties who do not trust one another are conducted safely. In order to do so, the source code in which the smart contract is written is compiled into bytecode and deployed to all nodes on the blockchain for execution. When a DApp is configured properly, it sends a message or transaction to a function of the corresponding smart contract. To do so, it needs the ABI (Application Binary Interface) to properly format the message and digitally sign it for submission. When the message is received by a node on the network, it is replicated to all the other nodes on the network for execution.

The above smart contract approach is designed for untrusted public networks. Replication helps ensure authenticity and agreement over untrusted networks, but it comes at a cost; if there are 1,000 nodes on the network, a single DApp's smart contract function is run 1,000 times each time it is requested. The slowest node on the blockchain sets the network's maximum execution speed, and the more logic incorporated in smart contracts, the slower the network executes. The performance consequences for business-to-business scenarios in trusted or semi-trusted environments can be devastating. However, this issue can be addressed by adapting a simple business application development model to smart contracts; this integration is done by using Cryptlets.

To get a better understanding of what Cryptlets are and how this process works, a division of the smart contract into its components is needed. These components are the properties (static and variable), the logic, and the ledger. Each of these components

can be mapped directly into technical concepts. Properties represent a data schema, logic represents code, and the ledger corresponds to a database. Once each of these components is defined, it can be deployed in environments that are optimised for its function.

With Cryptlets, once we separate the data and ledger from the logic, we can create a platform for the logic to run optimally. The contract's logic is packaged into a "cryptlet," which is a block or blocks of code that run inside a container, inside a fabric. These cyphers can be run on a different computer or in the cloud rather than on the actual nodes, and as a result, they do not need to be executed by every node on the network. Because they operate in a safe computing environment and have the cryptographic primitives needed to interact directly with blockchains, cryptlets can extend smart contracts outside of the blockchain while still maintaining the same level of security.

2.3 Which of the problems will be addressed?

The main elements that make smart contracts beneficial are that, like the blockchain, they are immutable, meaning that once a smart contract is implemented there is no way to change or "fool" it. However, a problem can arise if there is a bug (error) in the code, and this can only be fixed by creating a new contract and asking members to use it instead of the previous one. In addition, a smart contract is characterized by speed, efficiency and accuracy, as when a condition is met, the contract is executed immediately. Since smart contracts are digital and automated, there is no paperwork to handle and no time is wasted correcting errors that often occur when filling out forms manually.

Furthermore, because no third party is engaged in the distribution of smart contracts, there are no disparities, and because the encrypted records of transactions are shared amongst participants, there is no need to question if the information has been altered for personal advantage.

As already mentioned, smart contracts are commonly used in the blockchain, the blockchain transaction records are in an encrypted form, which makes it almost impossible to break them. Therefore, as each record is linked to the previous and

subsequent records in a distributed ledger, hackers would have to change the entire chain in order to be able to change a single record so, the security of smart contracts is unquestionable.

Finally, by using smart contracts instead of traditional contracts, both time and money are saved. It is widely known how time-consuming and costly it is to write and submit a contract due to the many intermediaries and fees that have to be paid. Smart contracts eliminate the need for intermediaries to execute transactions and hence the time delays and fees.

Smart contracts are usually found in three different forms:

i. Legal Contracts

These smart contracts were developed to streamline the legal system. They may be used for banking, real estate, and international commerce transactions and guarantee compliance to regulatory guidelines. While the existing legal system lacks the necessary framework to permit completely autonomous blockchain-based contracts, more and more of these will be used as laws and structures advance.

ii. Decentralized Autonomous Organizations (DAOs)

A decentralized autonomous organization (DAO) is a new type of legal structure with no central governing body and members who share the same purpose of acting in the best interests of the entity. A key aspect of DAOs are smart contracts, as they are designed for blockchain communities in which users have to follow the rules of the code. DAOs are open source and are most commonly used by crowdfunding platforms. Many smart contracts are used to manage, monitor and regulate community participation, while also ensuring their support.

iii. Application Logic Contracts (ALCs)

These contracts are at the heart of the internet of things (IoT). They are application-specific codes that interact with other applications on the same blockchain. They are

used to establish and confirm IoT device connectivity, combining IoT and blockchain technology. Every multipurpose smart contract will have a management software, which will comprise ALCs.

2.4 Real life implementations with details.

As explained above, smart contracts can be used in different situations, which vary depending on where companies apply them. Examples of such cases are:

i. Digital Identity

Digital Identity is one of the most apparent smart contracts use cases. Individual identity is one of a person's most valuable possessions. It is made up of reputation, data, and digital assets. If handled correctly, a person's digital identity can open up new doors for them. Currently the internet allows you to connect too many services, while inadvertently sharing your identity with organizations that are linked to your identity ownership. Smart contracts can allow counterparties to learn about an individual without knowing their true identity or validate transactions in this case. This seamless KYC¹ can help improve interoperability, robustness and security.

ii. Cross Border Payments

Smart contracts have the potential to change trade finance. There is no doubt that the use of a letter of credit can help in the international transfer of goods and the initiation of trade payments. It is clear that the adoption of smart contracts will increase the liquidity of financial assets, thereby enhancing the economic efficiency of suppliers, buyers and institutions.

¹ KYC means Know Your Customer

For smart contracts to work in trade finance, particularly in cross-border payments and international trade, an industry standard needs to be identified and implemented.

It can undoubtedly address legal issues and provide a better approach to resolving conflicts between parties if properly integrated.

iii. Loans and Mortgages

In addition, smart contracts can help to improve financial services such as mortgages and loans, as they can connect the parties and guarantee the smooth running of the whole project. Besides, it provides an error-free method that will save money and time since third parties involvement is not needed. For instance, a smart contract created to manage a mortgage can manage it by recording payments in the blockchain network and releasing the property when the loan is fully repaid. Because they replace manual tasks, transaction processing time is reduced and operations are optimized. This feature can also support the verification of tax returns for loan applications thus speeding up the approval process.

iv. Real Estate

In a blockchain, real-world assets can be tokenized and then smart contracts can be implemented to facilitate property transfers from buyers to sellers. Specifically, smart contracts allow sellers to break down their properties into fractions and offer buyers rights to them, which can be very beneficial for anyone looking to enter the real estate market and make small investments.

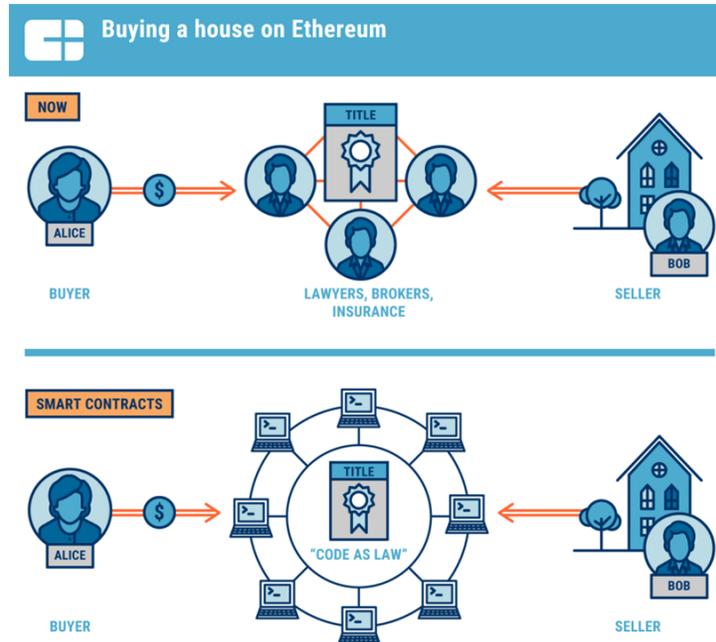


Figure 1: Picture source: cbinsights.com – How Blockchain Technology Could Disrupt Real Estate

v. Healthcare

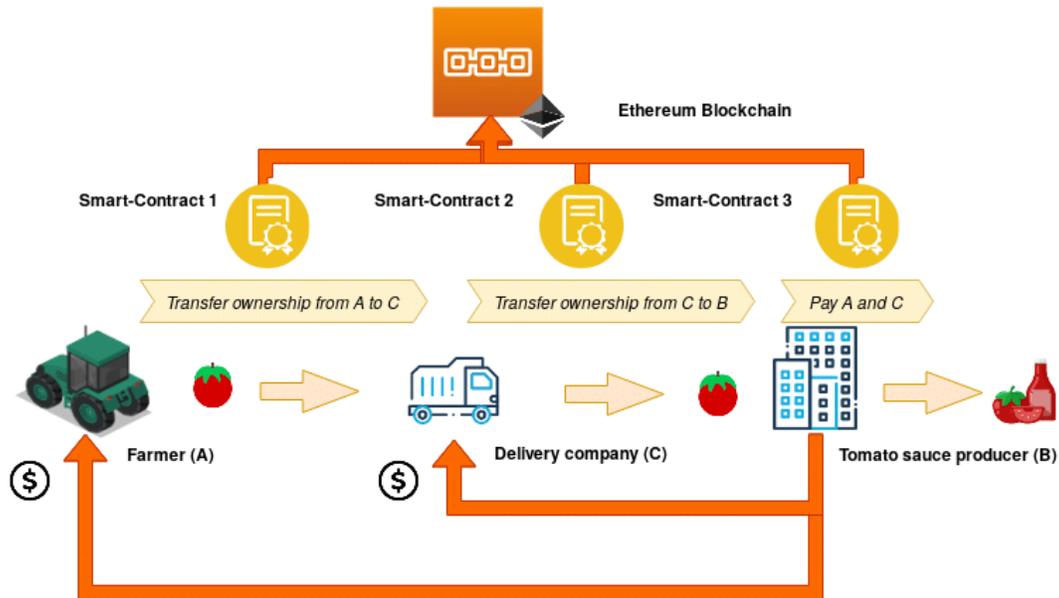
When patients' medical records are stored on a blockchain, then, with the use of smart contracts, the approval process for procedures could be automatically implemented. Finding a matching donor for an organ transplant across the globe, determining whether a patient's insurance company will cover a procedure, or ensuring cross-institutional visibility of data from clinical trials are just a few examples of what smart contracts can do.

vi. Supply chain

Supply chains can be improved by incorporating smart contracts, since they automate many procedures such as payments, shipments, and product management as well as to record both payments and status changes. They can also, through IoT devices and a blockchain oracles, notify managers or supervisors of various issues.

² Picture retrieved from: <https://www.cbinsights.com/research/blockchain-real-estate-disruption/>

Example of the application of Smart Contracts in the supply chain:



3

Figure 2: Application of Smart Contracts to traceability in the food-chain domain

Source researchgate available via license: [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/)

vii. NFT

The NFTs' value keeps increasing over recent years and, since they are non-feasible assets, they couldn't exist without smart contracts. Minting and transferring of ownership are secure through these contracts. Some of them can also be customized in terms of usefulness or appearance. For instance, an NFT may be programmed to change shape if certain conditions are satisfied, as well as programmed to alter the background using blockchain oracles. Finally, NFT developers may designate particular asset royalties when minting a new token, much like in the real estate industry.

2.5 Proposed resources

³ Picture retrieved from:

https://www.researchgate.net/publication/343232489_Combining_Blockchain_and_IoT_Food-Chain_Traceability_and_Beyond

Once a better understanding of the Smart Contracts methodology and the different applications are in place, all that remains is to explain how these contracts can be exploited within a company or by any stakeholder.

Following, some resources that can advise you on how to proceed from here.

- ANATOMY OF A SMART CONTRACT:

<https://www.blockchain-expo.com/2017/02/blockchain/anatomy-smart-contract/>

The historical context of the development of smart contracts and a strategy to address the new requirements while maintaining the advantages of the original Ethereum implementation are presented above.

- How to Create a Blockchain Smart Contract for Enterprise:

<https://www.devteam.space/blog/how-to-create-a-blockchain-smart-contract/>

All steps to implement blockchain-based smart contracts in an enterprise are explained.

- Smart Contracts Market:

<https://www.verifiedmarketresearch.com/product/smart-contracts-market/>

A broader analysis of the smart contracts market with complete insights into every aspect of this market (by blockchain platform, technology, end-user, geography, etc.)

- Top 6 smart contract platforms: a deep dive:

<https://www.itransition.com/blog/smart-contract-platforms>

The advantages and disadvantages of the 6 leading smart contract platforms are presented to give the user a better perspective when choosing the one that best meets their needs.

- Smart Contracts: <https://hedera.com/learning/smart-contracts>
A fully updated collection of different articles explaining smart contracts from theory to implementation and in different scenarios.
- Best Courses for Learn How to Create Smart Contract: <https://medium.com/javarevisited/best-courses-for-learn-how-to-create-smart-contract-2c9141ba2be9>
Various courses on smart contracts are listed and explained to deepen your knowledge.
- Introduction to Smart Contracts: <https://ethereum.org/en/developers/docs/smart-contracts/>
Through Ethereum's site a combination of smart contract explanation and visual presentation of the code behind them are been presented.
- Clean Contracts- a guide on smart contracts patterns & practices: <https://www.useweb3.xyz/guides/clean-contracts>
An overview of different patterns, techniques, and concepts for reducing the risks associated with smart contracts and blockchain technologies.
- Ethereum Smart Contract Security Best Practices: <https://consensys.github.io/smart-contract-best-practices/>
For intermediate Solidity programmers, this document offers a foundational understanding of security issues.
- Best Practices for Smart Contract Development:

<https://yos.io/2019/11/10/smart-contract-development-best-practices/>

A guide for smart contracts developers or experienced Solidity developers.

3. Knowledge Assessment

Question 1 true/false): What are smart contracts?

Smart contracts are self-executive contracts that are written in code and stored on a blockchain

[True]

[generic feedback]: Typically, are used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without any intermediary's involvement or time loss. They can also automate a workflow, triggering the next action when conditions are met.

Question 2 (multiple answers correct): For what reason can an enterprise use smart contracts for?

- A. [Provide identity management]
- B. [Improve financial transactions]
- C. [Streamline supply chains]
- D. **[All of the above]**

[generic feedback]: There are multiple reasons for integrating smart contracts into business operations, so depending on what you want to achieve, you can modify the smart contract accordingly to improve efficiency through step automation.

Question 3 (matching): Match the terms with their definitions.

Blockchain-based smart contracts: Self-executed statements written into code on a blockchain that are implemented when certain conditions are met.

Legal smart contracts: These smart contracts were developed to streamline the legal system.

DAOs (Decentralized Autonomous Organizations): A legal structure with no central governing body and members who share the same purpose of acting in the best interests of the entity.

Smart Application Logic Contracts: They are used to establish and confirm IoT device connectivity, combining IoT and blockchain technology.

Cyptlets: They host the logic of the smart contracts and are used for business-to-business scenarios in trusted or semi-trusted environments because they minimize the execution speed for the entire network.

[generic feedback]: There is an extensive terminology around smart contracts, as they are code-based contracts with a wide range of applications in different scenarios.

Question 4 (matching): Match the concepts with their explanations.

Smart contracts in real estate: Allow sellers to break down their properties into fractions and sell the rights to them.

Smart contracts on mortgages: Releasing the property once loans are fully repaid.

Smart contracts in the supply chain: Automate payments, shipments and product management.

Smart contracts in healthcare: Determine whether the insurance company will cover a procedure.

Smart contracts in digital identity: They contribute to improving interoperability, robustness and security.

[generic feedback]: When smart contracts are integrated into a blockchain used by businesses in various sectors, they can offer a range of new applications ranging from tracking/buying goods to supporting the medical sector and many others.

Question 5 (matching): Match the problems with their solutions.

- I want to buy a house but have to spend a lot of money and time on intermediaries: Smart contract between the buyer and the seller of the property.
- I want to donate money to a charity but I'm concerned my accounts may be hacked: Smart contracts use cryptography to secure transfers.
- We use smart contracts within the limits of our company, but their execution takes longer than expected: Embedding cryptles in the smart contract in order to increase execution speed.
- As a result of forgetting to pay the transport companies, I cannot get the restaurant's products delivered on time: Smart contracts keep a record of payments and can also automate payment processes.
- I want to purchase something online, but the payment has to be made in advance, so I am worried that I won't receive it: Smart contracts can be modified to release your payment once certain factors are met.
- [generic feedback]: The problems that smart contracts can address are numerous, as there is no limit to their applications. Most of the time the uses are related to the financial factor and product tracking.

4. Module Summary

When adapting blockchain technology to the standard operations of a business, the results can be revolutionary, as this technology can maximize results both in terms of reliability and security. Smart contracts are key to adapting blockchain to meet specific needs and business models. They are also able to serve as a safety net when exchanging

assets or support the automation of multiple internal flows depending on the data received.

5. References

- *Smart contracts*. Corporate Finance Institute. (2022, November 11). Retrieved November 25, 2022, from <https://corporatefinanceinstitute.com/resources/valuation/smart-contracts/>
- Gray, M. (2017, February 11). *Evolution of blockchain smart contracts and cryptlets*. Medium. Retrieved November 25, 2022, from <https://medium.com/@newgatemarleyg/evolution-of-blockchain-smart-contracts-and-cryptlets-cb22aa978434#:~:text=Contract%20Cryptlets%20define%20business%20logic,also%20in%20the%20Middle%20Tier.>
- Richencore. (2017, June 15). *Anatomy of a smart contract*. Blockchain Expo. Retrieved November 25, 2022, from <https://www.blockchain-expo.com/2017/02/blockchain/anatomy-smart-contract/>
- Peranzo, P. (2022, September 19). *Smart contracts in blockchain: Types, use cases & more*. Imaginovation. Retrieved November 25, 2022, from [https://imaginovation.net/blog/smart-contracts-in-blockchain/#:~:text=Application%20Logic%20Contracts%20\(ALC\)&text=These%20contracts%20contain%20an%20application,IoT\)%20merger%20with%20blockchain%20technology.](https://imaginovation.net/blog/smart-contracts-in-blockchain/#:~:text=Application%20Logic%20Contracts%20(ALC)&text=These%20contracts%20contain%20an%20application,IoT)%20merger%20with%20blockchain%20technology.)
- Lipton, A., & Levi, S. (2018, May 26). *An introduction to smart contracts and their potential and inherent limitations*. The Harvard Law School Forum on Corporate Governance. Retrieved November 25, 2022, from <https://corpgov.law.harvard.edu/2018/05/26/an-introduction-to-smart-contracts-and-their-potential-and-inherent-limitations/>
- *Smart contract challenges*. Hedera. (n.d.). Retrieved November 25, 2022, from <https://hedera.com/learning/smart-contracts/smart-contract-challenges>

- *What are smart contracts on Blockchain?* IBM. (n.d.). Retrieved November 25, 2022, from <https://www.ibm.com/topics/smart-contracts>
- Deltec Bank & Trust. (2022, February 15). *Smart contracts and Financial Services*. Deltec Bank & Trust. Retrieved November 25, 2022, from <https://www.deltecbank.com/2022/02/15/smart-contracts-and-financial-services/?locale=en#:~:text=Smart%20contracts%20are%20tamper%2Dresistant,to%20finance%20smart%20contracts%20bring.>
- Real World examples of smart contracts. Gemini. (n.d.). Retrieved November 25, 2022, from <https://www.gemini.com/cryptopedia/smart-contract-examples-smart-contract-use-cases>
- Goodness, U. (2022, April 6). *6 examples and use cases of smart contracts*. LogRocket Blog. Retrieved November 25, 2022, from <https://blog.logrocket.com/examples-applications-smart-contracts/>
- Davies, A. (2022, August 23). *How to create a blockchain smart contract for Enterprise*. DevTeam.Space. Retrieved November 25, 2022, from <https://www.devteam.space/blog/how-to-create-a-blockchain-smart-contract/>
- Smart contract platforms: A comprehensive selection guide. Smart Contract Platforms: a Comprehensive Selection Guide. (n.d.). Retrieved November 25, 2022, from <https://www.itransition.com/blog/smart-contract-platforms>