BRATISLAVA
LAW
REVIEW

PUBLISHED BY
THE FACULTY OF LAW,
COMENIUS UNIVERSITY
BRATISLAVA

ISSN (print): 2585-7088
ISSN (electronic): 2644-6359

A FRAMEWORK FOR EFFECTIVE SMART CONTRACTING /

Ioana Vasiu, Lucian Vasiu

Prof. Dr. Ioana Vasiu
Faculty of Law
Babeş-Bolyai University
11 Avram Iancu Street,
400089, Cluj-Napoca,
Romania
ioana.vasiu@law.ubbcluj.ro
ORCID: 0000-0001-6122-6124

Lucian Vasiu, Ph.D., MBA
Independent Computer Scientist,
Information Systems Security and
e-Business Expert.
lvcianvs@yahoo.com
ORCID: 0009-0003-0256-933X

Abstract: *Smart contracts are event-driven computer programs used to automatically execute all or parts of the agreements between two or more entities, pursuant to their specifications. The self-executing and self-enforcing attributes of smart contracts present numerous potential benefits, such as cost efficiency, accuracy, and reliability, as well as the potential to support several sustainable development goals. Smart contracts can be very efficient in many sectors, with important automation, procurement, financial, and other supply chain management features. For this study, a systematic literature review was performed, with a view to assessing, synthesizing, and critique the current state of legal and security aspects of smart contracts. The analysis of publications and reports gathered allowed the identification and mapping of the most relevant aspects and revealed numerous issues and vulnerabilities associated with the use of this technology. This paper provides the following contributions: the study and organization of a large corpus of relevant publications; the review of smart contract definitions, from several perspectives; an outline of smart contract characteristics; a framework for effective smart contracting, addressing legal and security issues and proposing several improvements.*

Key words: *Smart Contract; Cybersecurity; Risk; Legal Enforceability*

Suggested citation:
Vasiu, I., Vasiu, L. (2023). A Framework for Effective Smart Contracting. *Bratislava Law Review*, 7(2), 107-122. https://doi.org/10.46282/blr.2023.7.2.511

Submitted: 06 July 2023
Accepted: 13 December 2023
Published: 29 December 2023

1. INTRODUCTION

Sophisticated technologies increasingly impact the traditional business methods and the way in which various transactions are conducted. Smart contracts, for instance, are regarded as a key component of the fourth industrial revolution, with a major potential for numerous domains (Lin et al., 2022).

Numerous organizations are already embracing this technology, due to the significant potential benefits, such as reduced transaction costs; increased reliability; enhanced forms of collaboration or enforcement protocols; and improved sustainability (as contracts eliminate the need to use paper). Moreover, according to a survey cited in the Data Act (2022), 79% of the respondents regard smart contracts, in the co-generated data on the Internet of Things (IoT) context, as an effective data access and use tool.

The potential use of smart contracting is very large, encompassing, for example, decentralized financial (DeFi) services (Makarov & Schoar, 2022; Khan et al., 2021;

Zetzsche, Arner, & Buckley, 2020); construction industry (Ye, Zeng, & König, 2022); energy industry (Mishra et al., 2022); supply chains (Groschopf et al., 2021; Chang et al., 2019); crypto-assets exchange (In Re Bibox Group Holdings Ltd. Decs. Litig., 2021); non-fungible tokens trading (NFTs) (Hermès International and Hermes of Paris v. Rothschild, 2022); real estate ownership and transactions registration (Stefanović et al., 2022); prediction market (Kushwaha et al., 2022); loan industry (Symbiont.io v. Ipreo Holdings, 2021); digital rights management; streamline of operations in the retail industry (e.g., allowing the creation and delivery of orders), insurance industry, and healthcare services processes (i.e., sharing patient information and automate insurance payments); etc. Furthermore, smart contracts can be very instrumental in supporting several the United Nations sustainable development goals (SDGs), such as, for instance, Goal 3 (health and well-being), Goal 17 (partnerships, to coordinate and trace international aid transactions), Goal 8 (decent economic growth, through universal access to services such as banking or insurance), etc. (UNCTAD, 2021; Hughes et al., 2019).

Nonetheless, alongside the significant promises of smart contracts, there are several major conceptualization, implementation, and execution challenges (Zheng et al., 2020). These challenges are related to important legal and security requirements and can pose significant problems in practice. Moreover, the conventional software engineering process models are not fully adequate for the smart contracting environment, as these models do not adequately account for aspects such as the immutability of smart contracts upon deployment, assuming that modifications can be made easily via upgrades upon the software release (Sillaber et al., 2021). Therefore, the effective use of smart contracts depends on how well these issues or challenges are addressed.

There is a large corpus of relevant smart contracts literature. For instance, the Scopus search on <"smart contract" AND legal> displayed 572 documents, while the <"smart contract" AND security> search resulted in 5,030 documents; the Clarivate <"smart contract" AND legal> search displayed 184 results; the IEEE Xplore search on <"smart contract" AND "security issues"> produced 103 results; the Google Scholar search on <"smart contract" AND "legal issues"> displayed about 3,010 results. The literature discusses numerous related topics in detail, from technical, legal, or practical perspectives (Madine et al., 2023; Sahoo et al., 2022; Wu et al., 2022; Zhang et al., 2022; Barboni et al., 2022; Hewa et al., 2022; Ghodoosi, 2021; Reyes, 2020; United Nations/CEFACT, 2020; Manupati et al., 2020; Hasting, 2020; Singh et al., 2020; Fairfield, 2014).

This paper is structured as follows. The next section presents the research methodology, questions, and contributions. Section 3 contains two subsections, a review of smart contract definitions and a description of the essential characteristics of smart contracts, respectively. Section 4 analyses legal challenges related to smart contracting. Section 5 proposes a framework for effective smart contracting. The framework is understood as a "conceptional structure," an "openwork," or a "skeletal support used as the basis for something being constructed" (Webster Dictionary, 2023; The Free Dictionary, 2023). Finally, the paper draws the conclusion.

2. RESEARCH METHODOLOGY, QUESTIONS, AND CONTRIBUTIONS

This paper is based on an extensive literature review, from a variety of fields. As Snyder (2019) remarks, through the integration of findings and perspectives from numerous empirical findings, a literature review has the potential to better address research questions. Literature review is considered "an excellent way of synthesizing research findings to show evidence on a meta-level and to uncover areas in which more research is needed, which is a critical component of creating theoretical frameworks" (Snyder, 2019, p. 333).

The aim of the systematic literature review was to assess, synthesize, and critique the current state of the aspects concerning legal smart contracts in practice. This research found a complex corpus of relevant publications. Initially, several keywords and search strings were determined, such as "smart contracts vulnerabilities;" <"smart contract" AND vulnerabilities>; <"smart contract" AND "legal aspects">; <"smart contract" AND "security issues">. The identification of the relevant materials was based on keywords and strings searches in bibliometric databases, such as Scopus, Clarivate, and IEEE Xplore, and the search engines (Google and Google Scholar). To narrow the results, additional search strings were developed. The highest attention was given to journal articles, international organizations documents, laws and legal bills, and legal cases.

The materials obtained were filtered, processed, and organized based on the quality and relevance assessment. The most relevant materials were analysed in-depth, with respect to the following aspects: categories, issues, and contributions. The study of the materials employed content analysis, to systematically identify the relevant issues.

The paper aims to answer the following four research questions:

- How are smart contracts defined?
- What are the main characteristics of smart contracts?
- What legal aspects must be observed in the creation and execution of smart contracts?
- What are the most concerning smart contract security vulnerabilities and risks?

The main contributions provided by this paper are as follows: the study and organization of a large corpus of relevant publications; the review of smart contract definitions, from several perspectives; an outline of smart contract characteristics; a framework regarding effective smart contracting, from a legal and a security perspective, with several proposed improvements.

3. SMART CONTRACTS

3.1 Definitions

Smart contracts are difficult to define as there are multiple types and deployment, integration, and execution possibilities. Another issue is the imprecise, often interchangeable, use of terms such as "smart contract," "algorithmic contract," "executable contracts," "smart contract code," "legally binding smart contracts," "smart legal contracts," etc. Several international organizations and states have adopted legislation regarding smart contracts (Ferreira, 2021).

International Organizations

According to the United Nations Commission on International Trade Law (2020: 8), the expression "smart contract" is a "misnomer," as it refers to a program that is neither a "contract" nor, in the artificial intelligence sense, "smart." Nevertheless, there are numerous definitions of smart contracts, from various perspectives (technical, legal, etc.). The United Nations Commission on International Trade Law (2022: 3) conceptualizes "smart contracts" as "instances of the use of automated systems to perform contracts."

The International Telecommunication Union (2019) defines "smart contract" as being a program, available on a distributed ledger system, which encodes rules for certain transactions, validated and triggered by certain conditions. The Chamber of Digital

Commerce (2018), defines "smart contract" as computer code, stored on a distributed ledger, which, when certain specified condition or conditions happen, runs automatically, in accord with pre-specified functions and writes the results into that distributed ledger.

United States

In **Tennessee** (TN Code § 47-10-201), "smart contract" is defined as "an event-driven computer program, which executes on an electronic, distributed, decentralized, shared, and replicated ledger that is used to automate transactions, including, but not limited to, transactions that:

- Take custody over and instruct transfer of assets on that ledger;
- Create and distribute electronic assets;
- Synchronize information; or
- Manage identity and user access to software applications."

Arizona (AZ Statute § 44-7061) and **North Dakota** (N.D. Cent. Code § 9-16-19) have identical definitions and define the "smart contracts" as an "event driven program, with state, which runs on a distributed, decentralized, shared and replicated ledger that can take custody over and instruct transfer of assets on that ledger."

In **Wyoming**, the Decentralized Autonomous Organization Supplement (17-31-102) defines "smart contract" means "an automated transaction, as defined in W.S. 40-21-102(a)(ii), or any substantially similar analogue, or code, script or programming language relying on a blockchain which may include taking custody of and transferring an asset, administrating membership interest votes with respect to a decentralized autonomous organization or issuing executable instructions for these actions, based on the occurrence or nonoccurrence of specified conditions."

European Union

In the **European Union**, the Data Act (2022) provides the definition of "smart contract" in Art. 2 (16): "computer program stored in an electronic ledger system wherein the outcome of the execution of the program is recorded on the electronic ledger."

In **Italy**, Art. 8-ter of Law No. 12/2019 defines "smart contract" as a computer program which works on distributed ledgers, and which can bind parties involved based on the previously defined effects by those parties.

In **Malta**, the Digital Innovation Authority Act (2018) defines "smart contract" as "a form of innovative technology arrangement consisting of: (a) a computer protocol; and, or (b) an agreement concluded wholly or partly in an electronic form, which is automatable and enforceable by execution of computer code, although some parts may require human input and control and which may be also enforceable by ordinary legal methods or by a mixture of both."

3.2 Characteristics of Smart Contracts

Advanced technologies allowed significant developments in automated contracts and transactions, extensively discussed in the literature. Surden (2012), for instance, proposes the classification of technologically advanced contracts into "data-oriented" and "computable" contracts, which allow the expression of contractual terms in computer-readable forms. A taxonomy for "algorithmic contracts" is proposed by Scholz (2017, p. 136): algorithmic contracts are distinguished by the role played by the algorithm involved (that is, tool or agent), by the tasks involved (such as gap-filling or negotiation), and, concerning negotiating algorithms, as "black box algorithm" or "clear box algorithm." Cohney & Hoffman (2020, p. 323) discuss "transactional scripts," defined as "a persistent piece of software residing on a public blockchain [...] executed as a part of an exchange, the code effectuates a consensus change to the state of a ledger."

Smart contracts represent "the mature end of the evolution of electronic agreements over several decades" (Webach & Cornell, 2017, p. 317). Smart contracts represent an attempt to improve the formation and enforcement of obligations, as Fairfield (2022, p. 84) observes, "the interface between human—natural—language and computer programs matters, and legal constructions of human encounters with automatic systems have profound legal significance."

Smart contracts can be regarded as predefined relationships, as actions of a party triggers actions of another party. Smart contracts must "be considered contracts because they are agent-generated mechanisms to shift rights and obligations" (Webach & Cornell, 2017, p. 338). However, while Reyes (2020, p. 991) regards smart contracts as "merely another step in the chronological development of technology that enables computable contracting," Werbach & Cornell (2017, p. 318) note that "while smart contracts can meet the doctrinal requirements of contract law, they serve a fundamentally different purpose."

In practice, smart contracts involve two or more parties agreeing to a set of rules and intended results, which are coded. The data required by the contract is fed into blockchains from external sources. Real-time data feeds are named "oracles," and can be software- or/and hardware-based. Oracles can be internal or external, a fact that induces the risk of malicious or wrong data, without possible recurses to application layer security protocols.

Smart contracts can take several forms. Smart contracts can be simple (for instance, for Bitcoin transactions) or complex (for instance, certain contracts running on the Ethereum blockchain), in certain cases involving multiple conditions or the use of other smart contracts and several parties. Based on the type of execution, smart contracts can be classified as known or pre-fixed execution and linked to certain events or conditions.

Raskin (2017, p. 310) distinguishes between strong and weak smart contracts, the former with prohibitive costs of revocation or modification, while the latter including contracts that can be altered upon execution with relative ease. From a legal perspective, on the other hand, the European Law Institute (2022, pp. 13-4) distinguishes four smart contract types: (i) "mere code," with no legal agreement;" (ii) "a tool to execute the legal agreement," with the legal agreement existing off-chain;" (iii) "a legally binding declaration of will, such as an offer or acceptance or constitute a legal agreement itself;" and (iv) "merged with the legal agreement," therefore existing both on-chain and off-chain.

Smart contract requirements are specified as statement properties. Distinct types of logic are employed to express the specifications of smart contracts. These logics include temporal, dynamic, deontic, and defeasible logics (to define rights, obligations, and exceptions) (Tolmach et al. 2021). However, it is difficult to code in a smart contract what happens when there are contract performance deficiencies, or when one party is in breach of the terms of the contract. This aspect is further complicated by the fact that the code development tool chain, according to Zou et al. (2021), is not strong enough.

The basic components of smart contracts are the properties (static and variable), the code (which describes the commitments), and the ledger. After consensus is reached by the parties involved, the contracts are validated and authenticated, then written to a block in the blockchain iteration.

According to their specifications, and depending, potentially on other smart contracts, part or the entire code/agreement is automatically executed (Loon v. Department of Treasury, 2023). The execution of smart contracts writes any resulting data into the distributed ledger. Smart contract must be deterministic (i.e., the output must be the same on all nodes executing the code). However, certain smart contracts receive data from other smart contracts, raising sequencing and synchronizing challenges.

Initially, smart contracts were represented in a low-level, assembly-like language (Gec et al., 2023, p. 6). Currently, smart contacts can be written in several languages, such as Solidity, Vyper, Rust, Yul, Java, JavaScript, Python, Scrypto, etc. There are also libraries of modular, reusable smart contracts. The source code is compiled and executed inside blockchains. Noteworthy, smart contract code is limited in size, due to the blockchain infrastructure constraints, usually having between a few dozen to hundreds of lines of code. There are several platforms on which smart contracts can be deployed, such as Ethereum, Binance Smart Chain, Solana, or Cardano.

Smart contract transactions are instructions signed cryptographically from parties' accounts. These can be regular transactions (transactions from one account to another), contract deployment transactions, and execution of contracts (which interacts with a deployed smart contract). However, private keys involved are vulnerable to malicious attacks, which can result in significant losses for victims (Rivelli v. Doe, 2022).

The life cycle of smart contracts includes the contract generation, which comprises parties' negotiations, the formulations of specifications, and the writing and verification of the code, the release, and the execution (Wu et al., 2022). Essential smart contract characteristics are related to the blockchain technology: cryptography-based; transparent; quasi real-time execution; independence from any centralized party.

While smart contracts are conceived as immutable, there are instances where periodic developments are encountered, for instance, to update or upgrade certain services or for achieving interoperability, through newer versions of the contract and the deactivation of the old contract.

Certain smart contracts can easily be placed under the contract doctrine; however, others require interpretive work, for the application of contract law. However, not every smart contract can be construed as "legal contract." Certain smart contracts can represent a part of a large contract, others can be used to automatically execute other contracts, and some may not be contracts at all. Furthermore, smart contracts can be part of hybrid contracts, which combine natural language and code: certain obligations being recorded in a natural language, while others being coded in computer form, deployed on a distributed ledger.

Smart contracts can be valid under the United Nations Convention on Contracts for the International Sale of Goods (CISG), as they can satisfy the offer and acceptance requirements (Duke, 2019). While smart contracts can be legally binding, that is not the case where they were not regarded by parties as having the implications of traditional contracts.

There are numerous rules and principles of contract law which apply to smart contracts. In general, smart contract can be construed as enforceable in each jurisdiction if they are acceptable under the relevant fraud statute (contracts cannot have illicit purposes) and comprised all the essential terms applicable to traditional contracts' life cycle (Woebbeking, 2019). These requirements regard the parties' identification, the offer, the acceptance, the consideration, competency, and capacity.

Contracts are governed by the law chosen by the parties, which, according to the Rome I Regulation, must be "made expressly or clearly demonstrated by the terms of the contract or the circumstances of the case" (Art. 3). Parties can opt for the law applicable to the whole or to part only of the contract. Moreover, in situations involving international parties, the Hague Principles on Choice of Law in International Commercial Contracts 2015 can be chosen as the choice of law.

With respect to data-sharing, according to the European Union Data Act, smart contracts must comply with several requirements, including robustness and access controls that preclude functional errors or manipulations by unauthorized entities; termination and interruption in a safe manner; archiving and continuity capabilities; and consistency with applicable data-sharing agreements.

4. LEGAL CHALLENGES

Smart contracts are different from traditional contracts as they include, alongside the terms agreed by the parties, terms that are mandatory for the execution of the smart contract. To be legally enforceable, smart contracts must comply with the requirements of contract law. According to Schwartz & Scott (2003, p. 543), contract law "has neither a complete descriptive theory, explaining what the law is, nor a complete normative theory, explaining what the law should be." The legal requirements of a contract vary by jurisdiction; however, they include agreement (all contract parties accept the terms as they are presented); consideration (something of value offered to all contract parties); certainty; completeness; intention of the parties to make the agreement legally enforceable; and formal requirements.

The first requirement in the process of forming legally binding smart contracts is the agreement, which comprises an offer and the acceptance of that offer. The offer is an expression of the willingness to observe and execute the contractual terms, once accepted by the other contract party. Smart contract acceptance effectively amounts to the assent to offer's terms or conditions (Raskin, 2017). This requirement is not significantly different compared with traditional contracts; however, unlike in the case of traditional contracts, smart contract acceptance comes through performance.

The enforceability of smart contracts is determined at state-level. Several states recognize explicitly the legal authority to use smart contracts in electronic transactions. In Tennessee, for instance, smart contracts can exist in commerce and "No contract relating to a transaction shall be denied legal effect, validity, or enforceability solely because that contract contains a smart contract term" (TN Code § 47-10-201). In North Dakota, for another example, contracts may "not be denied legal effect, validity, or enforceability solely because the contract contains a smart contract term" (N.D. Cent. Code § 9-16-19).

Nonetheless, smart contracts may raise various legal considerations and numerous issues with respect to contract law requirements. Some of the problems encountered regard semantic consistency and common consent (Tong et al., 2022). For instance, computer code makes it difficult to express or define the semantics of certain situations, numerous legal concepts being exceedingly difficult to formalized (for example, "reasonable," "adequate," "minimal," "good faith," etc.). Moreover, even where the smart contract code has been executed, that does not necessarily imply that the contract is legally compliant and complete. Further, smart contracts, written in programming languages, raise readability and verification challenges (that is, the source code of a smart contract and the compiled code, to avoid any differences).

Additionally, as Fairfield & Selvadurai (2022, p. 117) point out, the assumption that people or organizations intend whatever code do when executed on decentralized ledgers "does not fit well with the law of contract." In certain cases, the code may "not reflect the considered, consciously anticipated choices of their corporate users" (Scholz, 2017, p. 137). The practical execution of the code will depend on the circumstances at the contract execution time, which may result in situations which were not considered when the code was deployed, or which are not currently desirable or even legal. Consequently, the fact that the code has been executed does not necessarily mean that the contract is certain or complete from a legal perspective.

Challenges may also arise with respect to the fact that there is currently no straight forward way to modify or amend a smart contract, thus creating challenges regarding the certain for the parties involved. For instance, a major practical problem arises when a smart contract party discovers errors in the code after it has been executed. Moreover, smart contracts cannot get information about outside events as they cannot include HTTP requests. Additionally, certainty may also be a challenge, as code may include variables, function modifiers, and events, thus allowing numerous actions, as well as with respect to the termination of smart contracts.

Another important legal aspect regards the way the parties "sign" the code. In the context of smart contracts, this can be done by applying the digital signatures to the coded transactions. Given the many potential complex arrangements, automated assessment may be insufficiently coded. The risk that the jurisdiction involved is not clearly determined or considered is very real, and this can result in the impossibility of determining whether the contract is lawful or not, as this depends on the actual applicable legislation.

The problem is further exacerbated by the difficulty of ascribing actual locations to digital code or transactions, a challenge which demands thoughtful consideration, as this can negatively impact the use of smart contracts in cross-border transactions. This aspect raised the question whether the parties should "have smart contracts governed not by a specific country's laws, but by supranational law, or even by soft law principles, such as the UNIDROIT Principles of International Commercial Contracts (De En Goh, 2022, p. 35).

The problematic aspects also include contract performance, since contracts are often drafted in ways which allows levels of "discretion, open-endedness, or abstraction to allow flexibility given future uncertainty" (Surden, 2012, p. 633). Even more difficult, in practice, can be to determine and address the breach of a smart contract, for instance, in cases of failures to perform, or where the performance is defective or interrupted. If a security breach is due to computer code, rather than to the actions of malicious persons, there can still be liability for the breach. In fact, the European Union Data Act (2022), in Art. 30, regarding essential requirements for data sharing, stipulates that vendors of applications using smart contracts must effectively address aspects regarding robustness (i.e., ensure that smart contract "avoid functional errors and to withstand manipulation by third parties"); safe termination and interruption; (c) data archiving and continuity; and (d) access control. The Data Act also stresses the importance of adopting common specifications for smart contract interoperability.

Code-only contracts can raise numerous issues with respect to their content and execution. On the other hand, contracts that, prior to being coded, have been agreed in a natural language, present a lower smaller scope for interpretation or disputes. This is an important aspect, at least from an execution and remedies perspective.

Ballell (2019), for instance, raises the issue of remedies related to the use of automated systems to perform contracts. Indeed, a few problems may arise in the contract life cycle, which can be associated with a range of remedies. However, numerous questions can be asked as to what extent the current remedies approaches, where "reliance on formal remedies is less frequent in smart contracts" (DiMatteo & Poncibó, 2019, p. 823), are adapted to smart contracting and whether technology-enabled automatic remedies are warranted (including, but not limited to the reversal of the effects of the smart contract code performance).

Several other important legal obligations, from multiple regulators, may also come into play with the execution of smart contracts, such as:

- Consumer protection (Forbes, 2022; D'Onfro, 2020), which regards several rights on the data involved, as well as the possibility to cancel the contract, conformity with existing standards, clearly stated remedies, and potential losses, which could be incurred, following certain unfair terms (Durovic & Willett, 2023).
- Securities laws (Risley v. Universal Navigation, 2023);
- Processing of personal data and the protection of privacy in electronic communications (Wu et al., 2022; Wan et al., 2022; Robles, 2020);
- Protection of intellectual property rights, which can be complex and, sometimes, difficult to anticipate, as it may regard, on one hand, the algorithms, code, trade secrets, and/or patentable materials used for the development of the smart contracts (Kleiman v. Wright, 2020), as well as the rights of other parties that have ownership interests, and which may be affected by during the execution of smart contracts.
- Prevention of discrimination.

5. FRAMEWORK

5.1 Security Issues

Cyberattacks are on the increase, both in numbers and sophistication, effectively rendering computer information systems and transactions vulnerable to numerous threats and attacks (Vasiu & Vasiu, 2018). According to a Statista survey (2023), more than half of the respondents consider data security as being the most critical cybersecurity area.

Apart from the known security risks, blockchains and smart contracts, due to several specific factors, such as the decentralized approach, the need to access external data sources, and the vast amounts of data that can be difficult to synchronize, present additional attack opportunities, which can result in major losses for victims. For instance, smart contracts may depend on the execution of parts of other smart contracts, and this kind of situation could lead to synchronization or sequencing issues. Further, not encountered in the case of text-based contracts, smart contracts present the risk of code programming errors or that the execution is subverted, resulting in unwanted or unintentional transactions. As deployed smart contracts are irreversible, the security problems are difficult to address, situation which can result in significant irreversible losses.

A list of attacks that occurred since 2016 on smart contracts can be found in Chu et al. (2023). Further, there are functional or transactional risks, which regard, for example, the limitations or failures of the underlying blockchain. Successful base-layer network attacks, for instance, can lead to application layer failures.

This situation underlines the importance of the security aspect. Zou et al. (2021), for instance, found that there was a remarkably high emphasis on ensuring smart contracts' code security; however, also found that 71.6% of the conducted survey respondents agreed that it was difficult to guarantee the security of smart contacts during development. This is genuinely concerning, as the value of the assets that can be stolen can be extremely high (for instance, the Decentralized Autonomous Organization or the parity multi-sig wallet hacks), and extremely difficult to track and recover the assets involved.

Smart contracts are vulnerable to several threats or risks. Atzei, Bartoletti, & Cimoli (2017) distinguish three levels: language, virtual machine and blockchain. Smart contract vulnerabilities regard numerous causes or types, such as reentrant calls, unexpected inputs, or unexpected branch executions (Otoni et al., 2022). Several taxonomies were proposed for smart contract vulnerabilities (Atzei, Bartoletti, & Cimoli, 2017) and numerous publications discuss these vulnerabilities (Ethereum, 2023; Zhou, et al., 2022; López Vivar et al., 2021; Porambage et al., 2021; He et al., 2020). Kushwaha et al. (2022), for instance, categorizes Ethereum smart contract vulnerabilities into three "main root causes" and seventeen "sub-causes" categories.

The most concerning vulnerabilities and risks associated with smart contracts are as follows.

Reentrancy: A severe vulnerability, which occurs when a function is called repeatedly, before the execution of a function is completed, and, as the variables of the function do not get updated after each call, can create serious issues. Reentrancy can be a single function, when attackers control one function, called recursively to conduct the unauthorized activity, and cross-function, when several functions, with shared implications, are controlled by attackers. Several techniques can be used to prevent this type of vulnerability (Zhou et al., 2022).

Overflow: This happens when the size of the value exceeds the constraints for a data type (top or lower, causing overflow or underflow) (Sayeed, Marco-Gisbert, & Caira, 2020). Addition, multiplication, and division overflows are the most encountered overflows in smart contracts (Fei et al., 2022).

Block randomness: Refers to the fabrication of malicious miners of blocks that result in deviation from the normal outcome of the pseudo-random generator (Zheng et al., 2020).

Callstack depth: Regards the situations where external calls fail, due to the exceeding of the maximum call stack. These situations, if not addressed adequately, permit attackers the forcing of malicious output.

Timestamp dependency issues: Refers to situations when the block timestamp which triggers the execution of an operation, is compromised by malicious attackers.

Transaction ordering dependency: Occurs due to concurrent orders of execution, which can result in incorrect execution results when the transactions depend on each other.

Data withholding: Occurs when the producer publishes blocks without sharing the data used to build the block. In such situations, the full nodes cannot verify the updates correctness, thus giving to the malicious block proposers the possibility to subvert the protocol rules push invalid state transitions.

Access control problems: When inadequate authorization or authentication mechanisms are in place, attackers can corrupt the smart contract data and/or functions or sign unauthorized transactions.

Unchecked Request Vulnerability: Where data or addresses are called by external controls, attackers can arbitrarily specify addresses, functions, and parameters related to such external calls (Chu et al., 2023). In successful attacks, smart contracts would perform functions that where not considered by the developers, resulting in financial losses for the victims (Chu et al., 2023).

Denial of service: Results in the disruption of the execution of a smart contract, by reverting the call every time.

To address these risks, parties employ code testing, however, there are numerous challenges to this, such as the difficulty to consider all cases or scenarios; potential flaws in compilers and virtual machines; lack of guidance for testing; lack of tools to measure code testing; gas consumption; etc. (Zou et al. (2021). Further, the validation of smart contracts requires a deep understanding of laws and of the semantics and potential events of each smart contract, to produce a complete set of scenarios, which may not be easy to develop, in several cases. Additionally, smart contracts may interact with unverified, even potentially malicious outside code (Bräm et al., 2021), further compounding the problem.

5.2 Necessary Improvements

Smart contracts can be legally binding agreements, however, not necessarily always. Incomplete or inharmonious legal provisions create uncertainty regarding the legal requirements or enforceability of smart contracts, potentially reducing the opportunities or willingness to use them. Further, currently it is difficult to code numerous aspects, for instance, regarding dynamic-adjustments, remedies, or consideration of new events, which would trigger the execution of smart contracts. Therefore, there is a clear need to adopt and harmonize adequate legal provisions, and to establish smart contract development, testing, and review standards, considering smart contracts' characteristics, limits, and vulnerabilities.

Smart contract must be fully tested, from entry, logic, and termination functions to access rights, considering the number of calls that can be handled and unexpected or unintended behaviours and data input issues. There should be acceptable and enforceable ways which stipulate how the risks and the liabilities related to the execution of smart contracts are allocated between the parties involved, as well as arbitration clauses. At least in the case of complex contracts, the development of hybrid contracts will allow to specify, in a clear manner, essential components, such as governing law or necessary contract updates.

The standardization of data representation, processing, and verification procedures should also receive appropriate consideration, as it is imperative to ensure untampered, unaltered, timely, and trustworthy data input to smart contracts. The development of templates which match various requirements, thoroughly tested, adaptive and allowing for dynamic specifications or requirements. Certified auditing services for smart contracts should also receive adequate consideration.

Smart contracts must be easily amendable or upgradable and terminable. The upgrade or termination of smart contracts, however, must be done in a safe manner, to prevent any attacks, with all parties involved approving the update or termination before its initiation. Furthermore, in cases of smart contract termination, no contract functions should be callable, all access or execution rights revoked, and, as appropriate, make impossible the contract reinstate.

The analysis and protective measures related to the security of smart contracts is of paramount importance. Therefore, developers and owners must ensure strong identity verification and adequate technical measures, to prevent unauthorized data access or modification or use of smart contract functions and the availability and reliability of the services provided, and the authentication of data (from authorized users only). Wrong or malicious calls must be immediately detected and reported to administrators.

The development of fuzzing execution, which would allow for the mining of potential security vulnerabilities, with a view to minimizing the security risks, is an important consideration in this context. Further, the development of formal specifications, which will allow for complex verification, to demonstrate the absence of coding errors (since, once deployed on blockchains, it is almost impossible to make revisions).

To prevent malicious or involuntary executions of smart contracts, the execution must be conducted in a time-controlled manner, with the owner having control over contract termination and interruptions. All the nodes must follow security protocols compliant with the smart contract requirements.

Finally, as the development, the execution, and the disputes involving smart contracts can be highly specialized, it is necessary to train and certify software engineers, law, and security professionals, with respect to all the aspects involved.

6. CONCLUSION

The automated execution of agreements through smart contracts presents numerous potential benefits. The range of agreements is very vast, ranging from business transactions or certain rights transfers to assets swaps and other complex types of transactions. This technology can increase the efficiency, traceability, and transparency of transactions, as well as other benefits. However, while smart contracts may play a significant role in future transactions, the technology is not yet fully developed, with numerous issues which must be addressed adequately. These issues regard the interplay of the coding various semantics and potential scenarios, the legal, and the security aspects.

This paper provides a framework for effective smart contracting. Smart contracts still face significant challenges in practice and will gain broad acceptance only if the technology fully satisfies the legal and security requirements. This mandates numerous improvements to the current situation. The paper analysed the main legal and security issues and challenges and proposed several improvements. The paper allows for the identification of specific smart contract requirements, underlines the main problems, and offers a platform for comprehensive system requirements conceptualization and systems management.

This study’s multidisciplinary underpinnings facilitate the holistic understanding of the complex issues related to smart contracts and the findings can be useful for lawmakers, lawyers, researchers, and smart contract developers.

BIBLIOGRAPHY:

Ballell, T. R. D. L. H. (2019). Legal challenges of artificial intelligence: modelling the disruptive features of emerging technologies and assessing their possible legal impact. *Uniform Law Review,* 24(2), 302-314. DOI: https://doi.org/10.1093/ulr/unz018

Barboni, M., Morichetta, A., and Polini, A. (2022). Smart Contract Testing: Challenges and Opportunities. In: *2022 IEEE/ACM 5th International Workshop on Emerging Trends in Software Engineering for Blockchain* (WETSEB), 21-24. DOI: https://doi.org/10.1145/3528226.3528370

Bartoletti, M., and Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts. In Proc. of POST. Springer, 164–186.

Bräm, C., Eilers, M., Müller, P., Sierra, R., and Summers, A. J. (2021). Rich specifications for Ethereum smart contract verification. *Proceedings of the ACM on Programming Languages, 5(OOPSLA)*, 1-30. DOI: https://doi.org/10.1145/3485523

Chamber of Digital Commerce (2018). Smart Contracts: Is the Law Ready? *Chamber of Digital Commerce*. Available at: https://digitalchamber.org/smart-contracts-paper-press/ (accessed on 15 February 2023).

Chang, S. E., Chen, Y., and Lu, M. (2019). Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process. *Technological Forecasting & Social Change,* 144, 1–11. DOI: https://doi.org/10.1016/j.techfore.2019.03.015

Chu, H., Zhang, P., Dong, H., Xiao, Y., Ji, S., and Li, W. (2023). A survey on smart contract vulnerabilities: Data sources, detection and repair. *Information and Software Technology*, 159, article 107221. DOI: https://doi.org/10.1016/j.infsof.2023.107221

Cohney, S., and Hoffman, D.A. (2020). Transactional Scripts in Contract Stacks. 105 *Minnesota Law Review,* 105, 319-386. DOI: http://dx.doi.org/10.2139/ssrn.3523515

D'Onfro, D. (2020). Smart contracts and the illusion of automated enforcement. *Washington University Journal of Law & Policy,* 61, 173-192.

De En Goh, G. R. (2022). Smart contract disputes and public policy in the ASEAN+ 6 region. *Digital Law Journal,* 3(4), 32–70. DOI: https://doi.org/10.38044/2686-9136-2022-3-4-32-70

DiMatteo, L. A., and Poncibó, C. (2019). Quandary of Smart Contracts and Remedies: The Role of Contract Law and Self-Help Remedies. *European Review of Private Law,* 6, 805–824. DOI: https://doi.org/10.54648/erpl2018056

Duke, A. (2019). What Does the CISG Have to Say About Smart Contracts? A Legal Analysis. *Chicago Journal of International Law,* 20(1), 141-177.

Durovic, M., and Willett, C. (2023). A Legal Framework for Using Smart Contracts in Consumer Contracts: Machines as Servants, Not Masters. *Modern Law Review*. DOI: https://doi.org/10.1111/1468-2230.12817

Fairfield, J. A. (2014). Smart contracts, Bitcoin bots, and consumer protection. *Washington and Lee Law Review Online,* 71(2), 35-50.

Fairfield, J.A.T., and Selvadurai, N. (2022). Governing the Interface Between Natural and Formal Language in Smart Contracts. *UCLA Journal of Law & Technology,* 27, 79-118.

Fei, J., Chen, X., and Zhao, X. (2023). MSmart: Smart Contract Vulnerability Analysis and Improved Strategies Based on Smartcheck. *Applied Sciences,* 13(3), 1733. DOI: https://doi.org/10.3390/app13031733

Ferreira, A. (2021). Regulating smart contracts: Legal revolution or simply evolution? *Telecommunications Policy*, 45(2), article 102081. DOI: https://doi.org/10.1016/j.telpol.2020.102081

Forbes, L. (2022). Consumer Protection In the Face of Smart Contracts. *Loyola Consumer Law Review,* 34(1), 45-78.

Gec, S., Stankovski, V., Lavbič, D., and Kochovski, P. (2023). A Recommender System for Robust Smart Contract Template Classification. *Sensors*, 23(2), 639. DOI: https://doi.org/10.3390/s23020639

Ghodoosi, F. (2021). Contracting in the age of smart contracts. *Washington Law Review,* 96(1), 51-92. DOI: http://dx.doi.org/10.2139/ssrn.3449674

Groschopf, W., Dobrovnik, M., and Herneth, C. (2021). Smart contracts for sustainable supply chain management: Conceptual frameworks for supply chain maturity evaluation and smart contract sustainability assessment. *Frontiers in Blockchain,* 4, article 506436. DOI: https://doi.org/10.3389/fbloc.2021.506436

Hasting, R. (2020). Smart Contracts: Implications on Liability and Competence. *University of Miami Business Law Review,* 28(2), 358-381.

He, D., Deng, Z., Zhang, Y., Chan, S., Cheng, Y., and Guizani, N. (2020). Smart contract vulnerability analysis and security audit. *IEEE Network,* 34(5), 276-282. DOI: 10.1109/MNET.001.1900656

Hewa, T. M., Hu, Y., Liyanage, M., Kanhare, S. S., and Ylianttila, M. (2021). Survey on blockchain-based smart contracts: Technical aspects and future research. *IEEE Access, 9.*

Hughes, L., Dwivedi, Y.K., Misra, S.K., Rana, N.P., Raghavan, V., and Akella, V. (2019). Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda. *International Journal of Information Management,* 49, 114-129. DOI: https://doi.org/10.1016/j.ijinfomgt.2019.02.005

Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., and Bani-Hani, A. (2021). Blockchain smart contracts: Applications, challenges, and future trends. *Peer-to-peer Networking and Applications,* 14, 2901-2925. DOI: 10.1007/s12083-021-01127-0

Kushwaha, S. S., Joshi, S., Singh, D., Kaur, M., and Lee, H. N. (2022). Ethereum smart contract analysis tools: A systematic review. *IEEE Access, 10*. DOI:10.1109/ACCESS.2022.3169902

Lin, S. Y., Zhang, L., Li, J., Ji, L. L., and Sun, Y. (2022). A survey of application research based on blockchain smart contract. *Wireless Networks,* 28(2), 635-690. DOI:10.1007/s11276-021-02874-x

López Vivar, A.L., Sandoval Orozco, A.L., and García Villalba, L.J. (2021). A security framework for Ethereum smart contracts. *Computer Communications*, 172, 119–129. DOI: https://doi.org/10.1016/j.comcom.2021.03.008

Madine, M., Salah, K., Jayaraman, R., and Zemerly, J. (2023). NFTs for Open-Source and Commercial Software Licensing and Royalties. *IEEE Access, 11*. DOI:10.1109/ACCESS.2023.3239403

Makarov, I., and Schoar, A. (2022). Cryptocurrencies and decentralized finance (DeFi) (No. w30006). Cambridge, MA, U.S.A.: National Bureau of Economic Research. DOI: http://dx.doi.org/10.2139/ssrn.4104550

Manupati, V. K., Schoenherr, T., Ramkumar, M., Wagner, S. M., Pabba, S. K., and Singh, R. I. R. (2020). A blockchain-based approach for a multi-echelon sustainable supply chain. *International Journal of Production Research,* 58(7), 2222–2241. DOI: https://doi.org/10.1080/00207543.2019.1683248

Mishra, S., Crasta, C. J., Bordin, C., and Mateo-Fornés, J. (2022). Smart contract formation enabling energy-as-a-service in a virtual power plant. *International Journal of Energy Research*, 46(3*)*, 3272-3294.DOI: https://doi.org/10.1002/er.7381

Otoni, R., Marescotti, M., Alt, L. Eugster, P., Hyvärinen, A.E.J., and Sharygina, N. (2022). A Solicitous Approach to Smart Contract Verification, *ACM Transactions on Privacy and Security,* 26(2), 1-28. DOI: https://doi.org/10.1145/3564699

Porambage, P., Gür, G., Osorio, D.P.M., Liyanage, M., Gurtov, A., and Ylianttila, M. (2021). The Roadmap to 6G Security and Privacy. *IEEE Open Journal of the Communications Society,* vol. 2, 1094-1122. DOI: 10.1109/OJCOMS.2021.3078081

Raskin, M. (2017). The law and legality of smart contracts. 1 *Georgetown Law Technology Review,* 304, 305-341.

Reyes, C. L. (2020). A Unified Theory of Code-Connected Contracts. *Journal of Corporation Law,* 46, 981-1001.

Robles, T., Bordel, B., Alcarria, R., and Sánchez-de-Rivera, D. (2020). Enabling trustworthy personal data protection in eHealth and well-being services through privacy-by-design. *International Journal of Distributed Sensor Networks*, 16(5). DOI: https://doi.org/10.1177/1550147720912110

Sayeed, S., Marco-Gisbert, H., and Caira, T. (2020). Smart contract: Attacks and protections. *IEEE Access, 8*, 24416-24427. DOI:10.1109/ACCESS.2020.2970495

Scholz, L.H. (2017). Algorithmic contracts. *Stanford Technology Law Review*, 20(2), 128-169.

Schwartz, A., and Scott, R.E. (2003). Contract Theory and the Limits of Contract Law. *Yale Law Journal,* 113(3), 541-619. DOI: https://doi.org/10.2307/3657531

Sillaber, C., Waltl, B., Treiblmaier, H., Gallersdörfer, U., and Felderer, M. (2021). Laying the foundation for smart contract development: an integrated engineering process model. *Information Systems and e-Business Management,* 19, 863-882. DOI: 10.1007/s10257-020-00465-5

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, 104, 333-339. DOI: https://doi.org/10.1016/j.jbusres.2019.07.039

Statista (2023). Critical cybersecurity areas worldwide 2022-2023. Available at: https://www.statista.com/statistics/1292944/critical-cybersecurity-area-worldwide/ (accessed on 3 May 2023).

Stefanović, M., Pržulj, D., Ristić, S., Stefanović, D., and Nikolić, D. (2022). Smart Contract Application for Managing Land Administration System Transactions. *IEEE Access, 10*. DOI:10.1109/ACCESS.2022.3164444

Surden, H. (2012). Computable Contracts. *University of California Davis Law Review,* 46 (629), 629-700.

Tolmach, P., Li, Y., Lin, S. W., Liu, Y., and Li, Z. (2021). A survey of smart contract formal specification and verification. *ACM Computing Surveys (CSUR),* 54(7), 1-38. DOI: https://doi.org/10.1145/3464421

Tong, Y., Tan, W., Guo, J., Shen, B., Qin, P., and Zhuo, S. (2022). Smart Contract Generation Assisted by AI-Based Word Segmentation. *Applied Sciences,* 12(9), 4773. DOI: https://doi.org/10.3390/app12094773

Wan, Z., Zhou, Y., and Ren, K. (2022). zk-AuthFeed: Protecting Data Feed to Smart Contracts with Authenticated Zero Knowledge Proof. IEEE Transactions on Dependable and Secure Computing, (01), 1-1. DOI: https://doi.org/10.1109/TDSC.2022.3153084

Vasiu, I., and Vasiu, L. (2018). Cybersecurity as an essential sustainable economic development factor. *European Journal of Sustainable Development,* 7(4), 171-178. DOI:10.14207/ejsd.2018.v7n4p171

Werbach, K, and Cornell, N. (2017). Contracts Ex Machina. *Duke Law Journal,* 67(2), 313–382.

Woebbeking, M. K. (2019). The impact of smart contracts on traditional concepts of contract law. *JIPITEC*, 10, 105.

Wu, C., Xiong, J., Xiong, H., Zhao, Y., and Yi, W. (2022). A review on recent progress of smart contract in blockchain. *IEEE Access, 10*. DOI: 10.1109/ACCESS.2022.3174052

Ye, X., Zeng, N., and König, M. (2022). Systematic literature review on smart contracts in the construction industry: Potentials, benefits, and challenges. *Frontiers of Engineering Management,* 9(2), 196-213. DOI: https://doi.org/10.1007/s42524-022-0188-2

Zetzsche, D. A., Arner, D. W., and Buckley, R. P. (2020). Decentralized finance. *Journal of Financial Regulation,* 6(2), 172-203. DOI: https://doi.org/10.1093/jfr/fjaa010

Zhang, L., Wang, J., Wang, W., Jin, Z., Su, Y., and Chen, H. (2022). Smart contract vulnerability detection combined with multi-objective detection. *Computer Networks, 217.* DOI: https://doi.org/10.1016/j.comnet.2022.109289

Zheng, Z., Xie, S., Dai, H. N., Chen, W., Chen, X., Weng, J., and Imran, M. (2020). An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems,* 105, 475-491. DOI: https://doi.org/10.1016/j.future.2019.12.019

Zhou, H., Milani Fard, A., and Makanju, A. (2022). The state of Ethereum smart contracts security: vulnerabilities, countermeasures, and tool support. *Journal of Cybersecurity and Privacy,* 2(2), 358-378. DOI: https://doi.org/10.3390/jcp2020019

Zou, W., Lo, D., Kochhar, P. S., Le, X. B. D., Xia, X., Feng, Y., Zhenyu, C., and Xu, B. (2021). *IEEE Transactions on Software Engineering,* 47(10), 2084-2106. DOI: 10.1109/TSE.2019.2942301

European Commission (2022). Proposal for a Regulation of the European Parliament and of the Council on harmonised rules on fair access to and use of data (Data Act), COM(2022) 68 final.

European Law Institute (2022). ELI Principles on Blockchain Technology, Smart Contracts and Consumer Protection.

Ethereum (2023). https://ethereum.org

Hermès International and Hermes of Paris v. Rothschild, No. 22-cv-384 (JSR) (S.D.N.Y. May 18, 2022).

In Re Bibox Group Holdings Ltd. Secs. Litig., 534 F. Supp. 3d 326 (S.D.N.Y. 2021).

International Telecommunication Union (2019). Distributed Ledger Technology Terms and Definitions, Technical Specification FG DLT D1.1, 1 August 2019. Available at: www.itu.int/en/ITU-T/focusgroups/dlt/Documents/d11.pdf (accessed on 15 February 2023).

Kleiman v. Wright, No. 18-cv-80176-BLOOM/Reinhart (S.D. Fla. Sept. 21, 2020).

Risley v. Universal Navigation Inc., No. 22 Civ. 2780 (KPF) (S.D.N.Y. Aug. 29, 2023).

Rivelli v. Doe, Civil Action No. 22-2060 (MAS)(RLS) (D.N.Y. Apr. 11, 2022).

Symbiont.io v. Ipreo Holdings LLC, CA No. 2019-0407-JTL (Del. Ch. Aug. 13, 2021).

Free Dictionary (2023). https://www.thefreedictionary.com/framework

United Nations Commission on International Trade Law (2020). Legal issues related to the digital economy – artificial intelligence, A/CN.9/1012/Add.1. Available at: http://undocs.org/A/CN.9/1012 (accessed on 19.12.2023).

United Nations Commission on International Trade Law (2022). The use of artificial intelligence and automation in contracting. A/CN.9/WG.IV/WP.173.

United Nations Conference on Trade and Development (UNCTAD) (2021). Harnessing blockchain for sustainable development: prospects and challenges. Available at: https://unctad.org/publication/harnessing-blockchain-sustainable-development-prospects-and-challenges (accessed on 15 February 2023).

United Nations/CEFACT (2020). White paper blockchain in trade facilitation. Available at: http://www.unece.org/fileadmin/DAM/cefact/GuidanceMaterials/WhitePaperBlockchain.pdf (accessed on 15 February 2023).

Webster Dictionary (2023). https://www.merriam-webster.com/dictionary/framework

Wyoming Decentralized Autonomous Organization Supplement (2023). https://sos.wyo.gov/Forms/WyoBiz/DAO_Supplement.pdf.