Enhancing Financial Integrity:

An Approach to Combat Money Laundering on the Blockchain through AML and KYC processes

July 27, 2023

SMART
CONTRACTS
LAB

Abstract

Money laundering is the act of making funds from illegal means appear as if they originate from legal sources. To prevent this, there are Anti Money Laundering (AML) laws and Know Your Customer (KYC) processes. However, cryptocurrencies pose a significant challenge to these preventive measures as they offer an avenue to bypass such controls by leveraging their inherent anonymity and their limited effectiveness of AML prosecution.

To combat money laundering through blockchain, this paper proposes a solution called KYCrypto. The solution provides a holistic concept that links regulatory processes to a cryptocurrency. Further, the paper introduces a smart contract that corresponds with the Ethereum Request for Comment 20 (ERC-20) standard. The smart contract provides tokens for whitelisted customers and enables AML prosecution through its blocking function. The oracle is the bridge between off-chain and on-chain processes. Thus, KYCrypto combines the advantages of the traditional banking system with the potential of blockchain technology, leading to an approach to regulate the environment within the blockchain.

Table of contents

List of tables and figures

List of abbreviations

AML	Anti-Money Laundering
AMLA	Federal Act on Combating Money Laundering and Terrorist Financing, Anti-Money Laundering Act
AMLO-FINMA	Ordinance of the Swiss Financial Market Supervisory Authority on combating money laundering and terrorist financing in the terrorist financing in the financial sector
BCBS	Basel Committee on Banking Supervision
CDB 20	The Agreement on the Swiss banks' Code of Conduct regarding the Exercise of Due Diligence
CDD	Customer Due Diligence
CIP	Customer Identification Program
DLT	Distributed Ledger Technology
EDD	Enhanced Due Diligence
ERC-20	Ethereum Request for Comment 20
euroSIC	euro Swiss Interbank Clearing
FATF	Financial Action Task Force
FINMA	Swiss Financial Market Supervisory Authority
GDP	Gross Domestic Product
IPFS	InterPlanetary File System
KYC	Know Your Customer
KYCE	Know Your Customer Exchange
PEP	Politically Exposed Person
PRISMA	Preferred Reporting Items for Systematic reviews and Meta-Analyses
SBA	Swiss Bankers Association

SCL	Smart Contracts Lab
SIC	Swiss Interbank Clearing
SIX	Swiss Infrastructure and Exchange
SWIFT	Society for Worldwide Interbank Financial Telecommunication
TARGET2	Trans-European Automated Real-time Gross Settlement Express Transfer System
USD	U. S. Dollar
VASP	Virtual Asset Service Provider

1. Introduction

According to United Nations (2023), the global annual amount of laundered money is estimated to range from 2% to 5% of the Gross Domestic Product (GDP), equivalent to 800 billion U. S. Dollars (USD) to 2 trillion USD. Within this amount, approximately 8.6 billion USD can be attributed to cybercrime committed using cryptocurrencies (Chainalysis, 2022). Chainalysis (2022) reports that it is very likely that the actual amount laundered using cryptocurrencies is even higher, as this amount only captures cybercrime-related activities.

A recent study by Pratz et al. (2021) reveals that European banks spend approximately 19 billion euros each year to ensure compliance with Anti-Money Laundering (AML) and Know Your Customer (KYC) regulations. Ratnawat et al. (2022, p.1) define KYC as follows: "Know Your Customer (KYC) is a process by which financial institutions validate the identity of customers or businesses with whom they conduct businesses. The KYC procedure is carried out to prevent institutions from being used/exploited for unethical and illegal activities such as money laundering, whether intentionally or unintentionally." Furthermore, Ratnawat et al. (2022) explain that the level of these activities is increasing, which shows the need for KYC (Ratnawat et al., 2022). However, the KYC-related processes are labor-intensive and recurrent since each bank must perform its own process (Ratnawat et al., 2022). This leads to higher overhead expenses (Ratnawat et al., 2022). The major reasons for these increasing costs are the constant changes and updates made by regulators in this field, along with the varying national implementations (Pratz et al., 2021). To solve these problems, it is essential to find a new solution (Ratnawat et al., 2022).

Despite the banks' efforts to redesign and simplify AML and KYC processes, they are facing challenges in reducing costs and fines associated with AML and KYC-related compliance (Pratz et al., 2021). To address this, one potential approach is to leverage new technologies, such as blockchain, to make processes more efficient (Pratz et al., 2021). Therefore, the aim of this paper is to propose a blockchain-based solution that cuts compliance costs while enhancing the efficiency of AML and KYC-related processes.

Furthermore, the AML and KYC regulations have only recently been applied or about to be applied to cryptocurrencies (Mulhim, 2022). Depending on the jurisdiction and on the crypto exchange, the level of regulation can differ significantly (PXL Vision, n.d.). However, this partial lack of regulation is diametrically opposed to an increasing acceptance of cryptocurrencies. Therefore, this paper aims to present a possible solution to advance regulations regarding cryptocurrencies.

Ultimately, the objective is to present an approach to regulating cryptocurrencies by creating our own conceptual solution that integrates the existing regulatory processes into a cryptocurrency framework. An integral part of this concept is implementing a smart contract that maps the central mechanisms of this conceptual solution. To achieve this, we will examine the below-stated research questions within the context of our study.

- What are the business processes involved in account opening and payment processing within the conventional banking system and the blockchain-based system?
- What are the AML and KYC regulations from the perspective of Switzerland?
- What are the current research approaches regarding the integration of AML and KYC-related processes within blockchain technology?
- How can the principles of AML and KYC be applied to a cryptocurrency conceptually?
  - How can a smart contract be designed that maps the central mechanisms of this conceptual solution?
  - Can the integration of AML and KYC generate additional revenue streams for banks?

To address our research questions, we initially analyze the business processes involved in account opening and payment processing within both the conventional banking system and the blockchain-based system. This analysis provides valuable insights into the similarities and differences between the two systems. Subsequently, we explore the actions resulting from AML and KYC regulations from the perspective of Switzerland by considering the requirements and guidelines. Afterwards, we systematically review current research on how blockchain technology can be applied for AML and KYC purposes. Following that, we introduce our own concept, called KYCrypto concept, which ensures that AML and KYC-related processes are applied to our self-developed token, called KYCrypto. Then, we summarize and discuss our findings critically by connecting them to existing research approaches. Lastly, we provide a conclusion along with suggestions for further research areas.

2. Method

This chapter explains our approach to creating a concept and a functional smart contract as a possible solution for regulating cryptocurrency and preventing money laundering. Initially, we analyzed blockchain technology and regulatory topics within the Smart Contracts Lab (SCL) project. Following that, we narrowed our focus down to researching topics about account opening, transactions, AML, KYC, and exploring corresponding blockchain alternatives to create a concept and code a smart contract.

The research, conducted with Swisscovery, Google Scholar, company websites, legal provisions, and guidelines on regulations, provided us with the fundamental information for our concept and smart contract. The information about the traditional system gave us insights into the current processes that needed to be included in our concept and smart contract, while the information about blockchain technology helped us in integrating these traditional processes. However, we also found new issues that we did not overcome by conducting research but by critically debating our thoughts with each other.

The KYCrypto concept went through multiple rounds of development. A basic version was initially created as a draft. Subsequently, this was refined through multiple revisions by incorporating different findings of the conducted research. Additionally, we reviewed the existing literature by analyzing the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) systematic review of Malhotra et al. (2022). The insights gained from this literature review helped us in designing our KYCrypto concept.

The smart contract for our KYCrypto concept was written in the coding language solidity. Our starting point was the code from the former students of Blockchain Presence AG, Oliver Vertesi and Roman Willi (Vertesi, 2022; Willi, 2021). Based on this, we redesigned the code and implemented crucial functions to align the smart contract with our concept. To test the smart contract, a fake oracle was created in collaboration with Florian Rüegsegger.

3. Account opening and account management (off-chain)

Initially, we analyze the current business process relating to account opening and account management, as we regard it as important to gain an understanding of the principles which apply in the off-chain environment before creating a blockchain-based concept.

According to Kaufman (1992), one of the services offered by banks is account opening and management. Certain Swiss banks may impose a fee for these services, although the charges can vary depending on the institution and which account a customer needs (Mekuli, 2022).

3.1 Account opening

In general, individuals who are legally mature and legally able to act are, as a matter of fact, permitted to establish a bank account in Switzerland (Swiss Bankers Association [SBA], 2023). In Switzerland, the banks are obliged to adhere to foreign legal and regulatory requirements concerning cross-border transactions (SBA, 2023). SBA (2023) further clarifies that banks have the right to reject customers. For instance, a bank may opt not to engage in a business relationship with a Politically Exposed Person (PEP) if there are concerns about potential damage to their reputation (SBA, 2023). Similarly, if there are uncertainties as to the source of funds of a prospective customer, the bank can reject establishing a business relationship (SBA, 2023).

When there is a request to open a bank account, a contractual relationship is established between the bank and the customer for a specific period (Swiss Federal Court, 1998). It is of great importance that the prospective customer with whom the contract is about to be entered is identified by the bank (Derleder et al., 2017a). Furthermore, Derleder et al. (2017a) state that for identification purposes, banks are bound by regulations regarding AML. These regulations ensure that banks follow due diligence obligations and take action to combat money laundering activities (Derleder et al., 2017a).

3.1.1 Types of bank accounts

In Switzerland, there are multiple types of bank accounts, and each has its purpose, as mentioned by Mekuli (2022). As various types of accounts can be differentiated according to their purpose, it is important for customers to have a clear understanding of their specific needs when selecting the most appropriate account (Mekuli, 2022).

- **Current accounts** represent one of the uncomplicated options for establishing a bank account (Mekuli, 2022). The author states that current accounts offer customers the convenience of managing bill payments, saving funds, receiving salaries, and making cash withdrawals. Typically, current accounts come with a monthly fee, and there may be additional charges for cash withdrawals at automated teller machines (Mekuli, 2022).
- **Savings accounts** act as a perfect complement to current accounts (Union Bank of Switzerland [UBS], 2023b). These accounts are suitable for setting money aside while having financial flexibility (UBS, 2023b). The objective of savings accounts is to accumulate funds over extended periods (Mekuli, 2022). However, funds in savings accounts carry, as a matter of principle, a higher interest rate compared to current accounts (Mekuli, 2022).
- **Investment fund accounts** have the purpose of helping customers steadily accumulate wealth and attain their long-term investment goals (UBS, 2023a). Investment accounts provide the opportunity for greater returns compared to savings accounts (UBS, 2023a). However, investing carries the risk of potential loss, whilst the risks are negligible in a savings account (Mekuli, 2022).

3.1.2 Account opening process

In the following section, the process of account opening is analyzed. It is important to acknowledge that bank account opening policies and processes must reflect AML and countering the financing of terrorism obligations (Basel Committee on Banking Supervision [BCBS], 2015). According to BCBS (2015), banks need to assess their customers in a differentiated manner because certain customers, such as PEPs, who usually carry a higher risk, require a more thorough screening than ordinary customers (BCBS, 2015).

As highlighted in the guidance from the BCBS (2015), it is mandatory for banks to follow the Financial Action Task Force (FATF) standards when conducting customer identification. The FATF is an international organization that establishes global standards to combat AML (Financial Action Task Force [FATF], 2023). Additionally, banks need to be cautious when processing information since some sources of information are more likely to be fraudulent (FATF, 2023; BCBS, 2015). If there is any doubt about the authenticity of the information provided, further verification should be carried out through additional inquiries or other sources of information (BCBS, 2015).

For individuals, banks typically need to collect specific information, as outlined in table 1. The pieces of information listed below are required for identification purposes.

Table 1

Information related to the account opening

Minimum Information	**Potential additional information (based on risks)**
First and last name	Any other names (marital name, former legal name, or alias)
Complete permanent address	Professional address, post office box number. The email address and landline or mobile telephone numbers
Nationality, an official personal identification number	Resident status
Date and place of birth	Gender

Note. Adapted from *General guide to account opening* (p. 3), by Basel Committee on Banking Supervision (BCBS), 2015, Bank for International Settlements (https://www.bis.org/bcbs/publ/d331.pdf). Copyright by Bank for International Settlements.

When opening an account, which marks the beginning of a customer relationship, further pieces of information, so called key attributes, are required to develop an initial customer risk profile (BCBS, 2015). The key attributes needed are listed in table 2.

Table 2

Key attributes

Key attributes	Potential additional information (based on risks)
Occupation, public position held	Name of employer, where applicable
Income	Sources of customer's wealth
Expected use of the account: amount, number, type, purpose, and frequency of the transactions expected	Sources of funds passing through the account
Financial products or services requested by the customer	Destination of funds passing through the account

Note. Adapted from *General guide to account opening* (p. 4), by Basel Committee on Banking Supervision (BCBS), 2015, Bank for International Settlements (https://www.bis.org/bcbs/publ/d331.pdf). Copyright by Bank for International Settlements.

3.2 Account management

Account management involves a set of guidelines, processes, and responsibilities that encompasses activities like initiating, terminating, and making changes to bank accounts (Farley, 2018). To make sure everything runs smoothly and to minimize risks, implementing efficient business procedures and safeguards concerning bank account information can enhance the financial management of the organization as well as the accurate maintenance of bank account data (Farley, 2018). Therefore, account management includes recording, processing, and issuing customer and order-related data as well as providing advice and support to customers (Barron, n.d.). The purpose of account management is to build and maintain relationships with customers (Barron, n.d.).

3.2.1 Account management regulation

Banks must comply with regulatory requirements (Swiss Financial Market Supervisory Authority [*Eidgenössische Finanzmarktaufsicht*; FINMA], 2023). The terms of relationship with customers are defined by the general terms and conditions. Banks adhere to these terms and conditions, although they may vary slightly depending on the bank (FINMA, 2023).

If the Swiss Financial Market Supervisory Authority [*Eidgenössische Finanzmarktaufsicht*] (FINMA) approves the banks and their regulations, the bank can implement them (FINMA, 2023). As Zanzi (2022) states, account opening is a task that falls within account management. Consequently, account opening fees and account management fees are part of account opening, and these fees vary from bank to bank (Zanzi, 2022). However, the author mentions that the customer usually pays an annual fee for the account. A possible reason for this is that maintaining accounts incurs administrative costs for banks (Zanzi, 2022). In addition to the fees incurred when opening an account, closing an account may also cost (Zanzi, 2022). These fees can vary among banks, and some Swiss banks do not even require fees for this (Zanzi, 2022).

According to Derleder et al. (2017b), one of the many duties of a bank is issuing balance statements. These statements give the customers an overview of their current account and are issued on a regular basis (Derleder et al., 2017b). In the balance statement, the mutual claims arising during the period, including the interest of the banks, are reconciled (Derleder et al., 2017b).

3.2.2 Customer support in banks

Customer care and customer responsibility revolve around service production, with a focus on integrating the customer into the process (Bruhn & Georgi, 2006). To ensure this, banks focus on customer interactions, such as assisting the customer at the bank counter or during counseling sessions. The traditional individual customer care approach, where a customer advisor serves the customer to the full extent and handles all of their banking transactions, is nowadays only applied in private banking and large business settings (Bruhn & Georgi, 2006). Bruhn and Georgi (2006) state that technological advancements and cost pressures led to a shift away from the traditional approach of assigning a customer advisor to a customer. Instead, the banking transactions of the customers are now channeled through multiple channels, aiming for personalized support on specific queries and standardized channels for regular transactions, according to the authors. Moreover, Bruhn and Georgi (2006) mention that key account management can be considered as a special form of customer care involving the provision of dedicated attention and support to important customers of a company, particularly in the banking sector, it is applied in private banking and wholesale business contexts (Bruhn & Georgi, 2006).

As already implied, according to Bruhn and Georgi (2006), customers were treated in a similar way when it came to customer service. However, a significant shift occurred, leading banks to implement customer segmentation, i.e., dividing customers into groups based on criteria such as product needs and customer profitability (Bruhn & Georgi, 2006).

4. Blockchain-based account opening

As we presented the traditional account opening, we carry on with the analysis of the blockchain-based account opening. The insights gained from this chapter helped us to design our solution. According to Parra-Moyano and Ross (2017), the KYC due diligence process is considered outdated and costly, which is why they propose a new approach by using Distributed Technology (DLT). This should enhance not only the cost-effectiveness but also the level of customer satisfaction (Parra-Moyano & Ross, 2017).

This chapter provides insight into how the account opening process from banks can be implemented in the blockchain world.

4.1 Blockchain-based account opening

4.1.1 Implementing a blockchain-based account opening

The mobile banking sector has already reduced access barriers by enabling customers to perform banking activities remotely without going to bank branches. However, further improvements may be achieved by using blockchain technology (Marsh & Maniff, 2017).

According to Marsh and Maniff (2017), banks can potentially utilize DLT to acquire and store documentation for account openings, particularly for customers facing barriers such as inconvenient hours and location (Marsh & Maniff, 2017). The authors state that blockchain technology could optimize the process of providing documentation for opening bank accounts, improving convenience for customers, and potentially attracting additional customers (Marsh & Maniff, 2017). The authors mention that one of the key advantages of DLT lies in its decentralized method for finding consensus in relation to the information stored. Thereby, it ensures that there is only a single distributed version of the truth (Marsh & Maniff, 2017). New transactions are added to the ledger, eliminating the need for third-party reconciliation, but this is only possible if the ledger is immutable and transparent since those are two essential features to establish a trustworthy environment (Marsh & Maniff, 2017).

Parra-Moyano and Ross (2017) suggest that using DLT reduces costs associated with the KYC verification processes. The banks can work together more effectively and efficiently by making use of the public key, whereby they can communicate and share data with each other more easily

(Parra-Moyano & Ross, 2017). With this new blockchain technology-based process, KYC verification is required only once by the customer, resulting in reduced cost, enhanced transparency, and the preservation of system security and privacy (Parra-Moyano & Ross, 2017).

4.1.2 Understanding crypto wallets

Unlike the process of opening a savings or checking account at a conventional bank, the way a crypto wallet is created heavily depends on a trade-off between security and convenience (Bharadwaj, 2023; Crowson, 2023). The choice of a crypto wallet depends on the intended use of the coins (Crowson, 2023). Each crypto wallet consists of two keys, a private key and a public key, which are interrelated and form the foundation of the asymmetric encryption used in transaction processing (Blocktrade, 2021). The public key is shared with others, while the private key must be kept confidential by the holder of the corresponding crypto wallet (Blocktrade, 2021).

There are two types of crypto wallets, custodial and non-custodial (Crypto.com, 2023). In the case of custodial crypto wallets, the provider is responsible for safeguarding the private key (Crypto.com, 2023). Many of the providers of custodial crypto wallets are crypto exchanges and/or brokers (Wouters, 2021). Some providers offer this service for free (Schwarz, 2023). However, some charge a quarterly or annual fee for the service (Gemini, 2022; Swissquote, 2023). Opening a custodial crypto wallet often requires proof of identification (Bharadwaj, 2023; Blocktrade, 2021). Custodial wallets are typically software solutions connected directly to the internet (Bharadwaj, 2023).

In the case of non-custodial wallets, the account holders themselves are responsible for managing the private key, ensuring its safety and secrecy (Crowson, 2023). For security reasons, it is advisable to use a so-called cold wallet, which is offline and not connected to the internet (Crowson, 2023). Although hardware wallets usually come at a higher price, they are a one-time fee compared to the recurring fees of custodial wallets (Gemini, 2022; Schwarz, 2023; Swissquote, 2023).

When comparing custodial and non-custodial wallets, it is crucial to consider the contrasting aspects of security and comfort (Bharadwaj, 2023). Custodial wallets provide backup options to prevent the loss of the private key, but they may have security vulnerabilities (Crypto.com, 2023). On the other hand, non-custodial wallets, particularly cold wallets, mitigate the risk of theft by being offline and disconnected from the internet (Crowson, 2023).

4.1.3 Revolutionizing account opening: the blockchain advantage

A blockchain-based account opening offers an alternative to the conventional method. While traditional banks rely on outdated and costly processes, blockchain technology presents a cost-effective solution (Parra-Moyano & Ross, 2017). Soltani et al. (2018) indicate that KYC and customer onboarding conducted by conventional solutions are often slow, resulting in high costs and dependent on face-to-face interactions. In contrast, a blockchain-based account opening is much more convenient from a customer's perspective (Raskin & Yermack, 2018). However, despite its potential benefits, doubts could arise regarding the privacy aspect since a ledger is transparent (Marsh & Maniff, 2017).

5. Overview of conventional payment processing in Switzerland

Introducing traditional and blockchain-based account openings lays the foundation for financial transactions. This chapter aims to describe the conventional bank transfer and the transactions on blockchain, including the way payment systems work, including Swiss Interbank Clearing (SIC), euro Swiss Interbank Clearing (euroSIC), and Society for Worldwide Interbank Financial Telecommunication (SWIFT).

Payment processing has undergone significant changes in recent years due to digitalization (European Payments Council, 2021). The payment system in Switzerland consists of various systems. This section provides an overview of Swiss banks' payment initiation and processing methods.

5.1 Traditional digital payment process in Switzerland

From the outside, the payment process seems straightforward. However, it is a complex procedure that involves multiple parties. Banks' specific processes and methods to facilitate payments vary based on factors such as the participants involved, the type of transaction, and the specific banking regulations in place (McKinsey, n.d.). However, banks typically prioritize reducing operating costs and increasing reliability in their payment processes to meet customer expectations (Deloitte, n.d.).

In general, bank-to-bank transfers involve several steps. First, the customer typically provides the necessary information, such as the recipient's name, account number, and the amount to be transferred to the bank (Zürcher Kantonalbank, 2023). Subsequently, the bank verifies the customer's account balance to ensure sufficient funds are available for the payment (DeMarco, 2022). Throughout the payment process, banks employ security measures, such as encryption, multi-factor authentication, and other protocols (Brennen, 2021). Fees charged by banks vary depending on the payment method and the transaction type (UBS, 2023c). While domestic payments are generally inexpensive, for international payments, banks often charge higher fees (Expatica, 2023). The service provider typically charges a fee alongside a percentage of the transferred amount and the provided exchange rate (Expatica, 2023).

Switzerland offers a range of payment methods, including wire transfers, contactless payments, and popular digital payment options like Twint. While direct transfers with Twint are possible in real-time, this immediacy is simulated as the bank relies on advance payments (Swiss Infrastructure and Exchange [SIX], 2022). However, the introduction of real-time transfers is anticipated through the SIC payment system starting in 2024 (SIX, 2022). This advancement may enable customers to send and receive money within seconds, thereby enhancing the speed and convenience of transactions (Bhatt, n.d.).

5.2 Swiss interbank clearing system

The SIC system is one of the pioneering real-time gross settlement systems and serves as a domestic payment system used for bank transfers within Switzerland (Heller et al., 2000). As a subsidiary of the Swiss Infrastructure and Exchange (SIX) Group, SIX Interbank Clearing Ltd is responsible for operating this payment system for the Swiss national bank (Swiss National Bank, 2023). As an automated clearing and settlement system, the SIC system enables banks in Switzerland to efficiently process large volumes of transactions between accounts (SIX, 2023e).

Every transaction denominated in Swiss francs goes through the SIC system, encompassing a diverse range of financial transactions, including bank transfers, salary payments, securities trading, and credit card payments (SIX, 2023e). According to SIX (2023e), the SIC system processes payments in real-time, allowing submitting and receiving transactions around the clock. In addition, the Swiss financial infrastructure provider states that they employ digital signatures to validate the payment orders and undergo a thorough data verification to confirm the legitimacy of the data input. SIC securely settles payments and debits or credits participants' settlement accounts when sufficient funds are available (SIX, 2023e).

Furthermore, SIX (2023e) declares that participants have real-time access to monitor incoming and outgoing payments, allowing them to stay updated on their settlement account's status. The purpose of SIC is to enhance the reliability as well as efficiency of interbank funds transfers while mitigating associated risks for banks and the central bank on a daily basis (SIX, 2023e).

5.3 Euro Swiss interbank clearing and TARGET2 system

As a gateway to payments in euro, euroSIC allows Switzerland, as a non-European Union member state, to connect with European financial centers (SIX, 2023d). EuroSIC system, which SIX Group also operates, enables the exchange of payments in euro between banks in Switzerland and all financial institutions in the European Union and European economic area (SIX, 2023b). According to the SIX, euroSIC participants maintain a clearing account, which undergoes daily reconciliation via a current account at the Swiss Euro Clearing Bank. The Swiss Euro Clearing Bank acts as a supervisory entity and monitors euro payments within and outside Switzerland (Swiss Euro Clearing Bank, n.d.). The advantages of the euroSIC system include the fast processing of cross-border euro payments, the ability to process wholesale and retail payments, and cost savings due to the automated processing of payments (SIX, 2023d).

The euroSIC system in Switzerland is interconnected with the Trans-European Automated Real-time Gross Settlement Express Transfer System (TARGET2) (Peyton, 2016). Cross-border transactions in the European Union go through the TARGET2 system (European Central Bank [ECB], 2020). It is operated and managed by the eurosystem and settles cross-border euro transactions within the European Union (ECB, 2023). Participants in the TARGET2 system are central banks and commercial banks (ECB, 2020). This payment system facilitates the processing and settlement of payments by utilizing the funds in their respective accounts held at the central bank (ECB, 2020).

With TARGET2, transactions are executed in real-time, minimizing the risk of non-payments and delays (ECB, 2020). Furthermore, ECB (2020) declares that effective management of liquidity is crucial for ensuring the execution of payments in real-time gross settlement systems. To ensure liquidity, multiple beneficial elements, such as transaction limits, priority settings, payment schedules, consolidation, and liquidity monitoring, are incorporated into the TARGET2 system (ECB, 2020). These elements enable the system to effectively address liquidity-related challenges and ensure smooth payment processing (ECB, 2020).

Overall, the SIC system and the ECB system enable transactions to be processed quickly and efficiently, saving time and money (SIX, 2023a). It also provides a secure way to transfer data between banks, effectively mitigating the associated risk of intermediaries (SIX, 2023b). In addition, the SIC system helps in the stability of the banking system in Switzerland by ensuring that transactions are processed quickly and efficiently (SIX, 2023c). Furthermore, banks can apply SIC or euroSIC through different means, such as using a Messaging Gateway, accessing the payment system with a web portal, or leveraging the SWIFT network (SIX, 2023e).

5.4 SWIFT network

Traditional bank transfer processes typically involve multiple intermediaries, such as banks and clearinghouses. SWIFT is a well-established network facilitating communication between banks through predetermined messages and instructions (Scott & Zachariadis, 2013). It enables cross-border transfers in multiple currencies, including Swiss francs, and facilitates communication between financial institutions (Seth, 2023). However, SWIFT does not hold funds, manage accounts, or carry out clearing and settlement operations (Kowsmann & Talley, 2022). Once a payment has been initiated, it is necessary for it to be settled via a payment system, such as TARGET2 (Scott & Zachariadis, 2013).

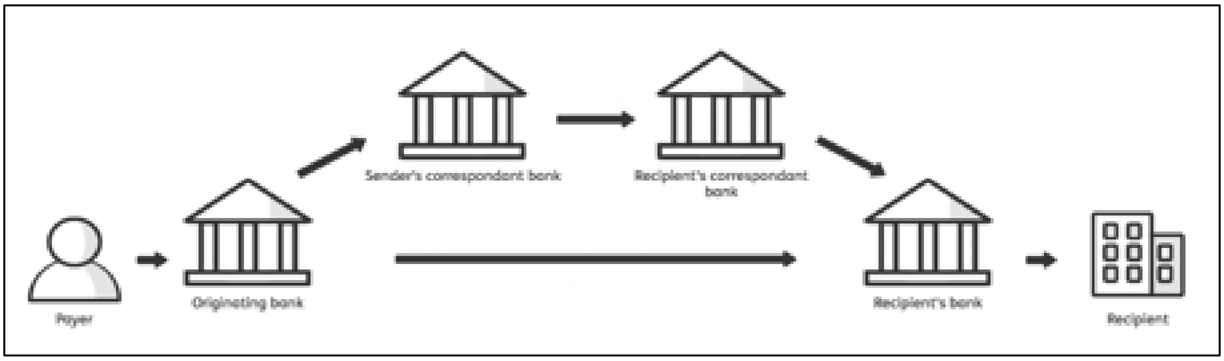
For identification purposes, each financial institution within the network has a unique SWIFT ID, known as a business identifier codes or SWIFT codes (Qiu et al., 2019). This SWIFT ID enables global identification and serves to identify each member of the network uniquely (Qiu et al., 2019). If two banks have an established relationship and hold a commercial account with each other, money can be transferred within minutes by a SWIFT message (Scott & Zachariadis, 2013).

For example, when Alice transfers 100 Swiss francs to Bob through her bank, the transaction process involves multiple steps (Acharya, 2021). To initiate a bank transfer through SWIFT, the sender provides the beneficiary's bank details, including the bank's SWIFT code, account number, and other relevant information. Alice's bank sends a SWIFT message to Bob's bank to initiate the transfer (Acharya, 2021). In this process, 100 Swiss francs are deducted from Alice's bank account, and Bob's bank credits the same amount to its commercial account (Acharya, 2021). Subsequently, the funds are added to Bob's personal bank account (Acharya, 2021).

Assuming Alice's and Bob's banks do not have a commercial bank account with each other, a third bank becomes involved as another intermediary that both banks have a commercial bank (Acharya, 2021). In such a scenario, the transaction becomes more complex. Alice's bank informs the intermediary bank to deduct 100 Swiss francs from its commercial account and credit the same amount to Bob's bank's commercial account (Acharya, 2021). Finally, Bob's bank adds the funds to Bob's personal account (Acharya, 2021).

Figure 1

SWIFT transaction



Note. Adapted from *MT103*, by trojanvilla, 2022 (https://trojanvilla.com/mt103/). Copyright by trojanvilla.

The involvement of additional intermediaries in the transaction process leads to specific implications. First, it results in higher fees and longer processing time because of the increased complexity (Arnold, 2018). SWIFT transfers can take a few business days to complete, depending on the participating banks and any intermediate correspondent banks involved (Qiu et al., 2019).

6. Blockchain transactions

After providing an overview of the traditional transaction process, this chapter takes a closer look at the transactions in the blockchain. The insights gained from this analysis help us to design our own solution, particularly in respect of transaction monitoring.

In the upcoming sub-chapters, the paper shows how blockchain-based transactions work, including using wallet addresses as a unique identifier and its features, such as pseudonymity. We also highlight the distinctions between coins and tokens, along with a comparison between traditional payment systems and blockchain-based systems.

6.1 Transactions on blockchain

For receiving and sending cryptocurrencies or other crypto assets, a wallet address is required. It acts as a unique identifier that stores private and public keys (Suratkar et al., 2020). The wallet address is generated from the public key by using a one-way mathematical function known as hashing (Di Pierro, 2017). In traditional banking, the equivalent counterpart to this unique identifier is the bank account number. The user can share the wallet address with others, and with this information, funds can be transferred (Suratkar et al., 2020). It is essential to emphasize that the public key and the wallet address are different from each other (Blocktrade, 2021).

Transactions on blockchain are decentralized, indicating that they are not controlled by a centralized intermediary or authority, such as a bank (Kaulartz & Heckmann, 2016). Beyond the public ledger, asymmetric encryption provides additional security (Simmons, 1979). Unlike symmetric encryption, the author emphasizes that the asymmetric encryption method utilizes distinct keys for encryption and decryption. The transmitter encrypts the transaction with the receiver's public key and decrypts it with the private key (Blocktrade, 2021). The private key is, as its name says, not public, and it is computationally infeasible to derive it from the public key, even if they are linked (Simmons, 1979).

While your bank or payment provider may charge transaction fees, gas fees serve as the equivalent counterpart on the ethereum network. Gas fees are paid to the node operators to compensate them for their work, as transactions on ethereum, or generally on the blockchain network, require no intermediaries (Roughgarden, 2020). The gas fees are dependent on the underlying asset, as each cryptocurrency has its own fee structure (Blocktrade, 2021). In case of ethereum, the gas fee is determined by the product of the gas limit and gas price (Roughgarden, 2020). The creator of the transaction, or the sender, sets the gas limit, which represents the maximum threshold for computational work respectively gas for the payment transaction (Roughgarden, 2020). The amount of gas required is determined by the complexity of the transfer (Blocktrade, 2021). The gas price reflects the sender's willingness to pay per unit of gas, which should be at least as high as the base fee, also known as the minimum price (Azouvi et al., 2023). The minimum pay varies based on network congestion and demand (Donmez & Karaivanov, 2022). According to the authors, in case the sender is willing to pay a high optional tip (the difference between the willingness to pay and the base fee), validators will prioritize the underlying transactions since their potential reward is higher than the others.

$$Gas\ fee = gas\ limit * gas\ price = gas\ limit * (base\ fee + optional\ tip)$$

As an example, if a transaction has a specified gas limit of 21'000 and the gas price is set at 0.0000002 ether, the resulting gas fee would be 0.0042 ether.

However, transactions on the blockchain are traceable, resulting in a transparent transaction history of crypto wallet addresses (Hayes, 2023). The identity behind the address cannot be determined, although the transaction history is accessible to third parties (Hayes, 2023). Thus, it is essential to note that transactions on the blockchain are considered pseudonymous rather than anonymous (Blocktrade, 2021).

6.2 Similarities and differences between coins and tokens

Coins and tokens are distinct from each other in various ways. Coins are digital assets that refer to a cryptocurrency and operate on their own dedicated blockchain networks (Nahar, 2022). As standalone currencies, the purpose of coins is to store or exchange value (Bots, 2022). Moreover, you can mine or stake coins for transaction validation and to secure a network (Nahar, 2022). Unlike tokens, they represent digital assets and operate on top of existing coins' blockchain networks (Bots, 2022). Their purpose is to leverage the underlying blockchain infrastructure to enable various functionalities, such as ownership representation, distribution of governance and voting rights etc. (Böhl, 2022).

The transaction of coins and tokens adheres to similar principles. All transactions are recorded on the native blockchain or on the ledger of that blockchain (Bots, 2022). However, there are also some differences in their transactional procedures. While coin transactions are executed on their native blockchain, token transactions typically engage with smart contracts (Nahar, 2022). Moreover, not all wallets support the storage and exchange of specific tokens (Crypto.com, 2022). Therefore, a compatible wallet embracing the specific token standards is essential to ensure seamless interaction with smart contracts.

6.3 Comparison between traditional and blockchain payment system

Transactions in traditional banking are reversible in the sense that transactions can be reversed without the consent of the parties involved, while blockchain transactions in cryptocurrencies are typically irreversible (Jaag & Bach, 2015). Blockchain transactions are generally irreversible once confirmed, relying on decentralized technology, and providing transparency as well as security (Heires, 2016).

The fact that the blockchain structure cannot be reversed means one must be cautious when it comes to transaction processes, ensuring that the protocols prevent human errors by developers and external attacks (Peters & Panayi, 2016). Transactions processed using the conventional payment system can be reversible, to some extent, since there are procedures and protocols to address issues like fraud, errors, or disputes, which may enable reversals or refunds for transactions (Jaag & Bach, 2015). The authors declare that transactions conducted on a blockchain are typically designed to be irreversible once confirmed, which shifts the risk from the receiver to the sender of the transaction. Every transaction recorded on the blockchain presents a permanent part of the transaction history, making it extremely difficult to reverse or alter (Heires, 2016). This immutability is a fundamental characteristic of blockchain technology (Heires, 2016).

While blockchain transactions are based on decentralized technology, SIC, euroSIC, and SWIFT rely on centralized systems operated by traditional financial institutions. Blockchains operate on globally distributed nodes, and transactions are validated through consensus mechanisms like proof-of-work or proof-of-stake (Akbar et al., 2021). In comparison, traditional systems manage and process transactions, relying on intermediaries as shown in chapter 5. The absence of a central authority enables peer-to-peer transactions and eliminates the need for intermediaries (Peters & Panayi, 2016). By eliminating intermediaries, blockchain technology can provide faster and cheaper payments (LinkedIn, n.d.).

Transactions processed through SIC, euroSIC, and SWIFT may take time to settle, especially in the case of international transfers involving multiple banks and their correspondent banks (LinkedIn, n.d.). In comparison, blockchain transactions, particularly in cryptocurrencies, can offer faster settlement times, especially for peer-to-peer transfers within the same blockchain network (Jaag & Bach, 2015). The transaction speed and associated fees can vary based on factors such as network congestion and chosen transaction fees (Donmez & Karaivanov, 2022).

7. AML and KYC within the regulatory frameworks

The previous chapters about account opening and transactions play a pivotal role in AML and KYC, particularly in the Customer Identification Program (CIP) and transaction monitoring aspects. Our main objective is to develop a holistic concept and design a smart contract that regulates cryptocurrencies. Before that, however, it is necessary to create an understanding of AML and KYC within the regulatory frameworks.

This chapter is divided into five sub-chapters. The first sub-chapter serves as an introduction, defining the terms money laundering and AML. The second sub-chapter analyzes the relevant regulatory frameworks and Swiss legislation about money laundering and AML. The third sub-chapter establishes the connection between these regulatory frameworks and the KYC process, covering topics such as the CIP, risks and a risk-based approach, Customer Due Diligence (CDD), ongoing monitoring, and KYC data storage. The fourth sub-chapter analyses AML prosecution that may arise from the KYC process when money laundering is suspected. Lastly, the fifth sub-chapter explores the state of AML and KYC in the context of virtual assets. A more detailed analysis of KYC and blockchain technology is provided in the literature review section.

7.1 Money laundering and anti-money laundering

According to the United Nations (2023), a substantial portion of the global GDP, between 2-5%, is subject to money laundering each year. This translates to 800 billion to 2 trillion USD (United Nations, 2023). Notably, the lower boundary of this estimate exceeds the GDP of Poland, which holds the 24th position in terms of global GDP rankings (International Monetary Fund, 2023). The upper boundary surpasses the GDP of Italy, the eleventh-largest economy globally (International Monetary Fund, 2023). Consequently, money laundering represents a concern due to the considerable volume of money laundered on an annual basis.

Upon quantifying the amount of money laundering, it is essential to state an accurate definition of money laundering. In the Swiss Criminal Code, the act of money laundering is defined under art. 305 para. 1 as follows: “Any person who carries out an act that is aimed at frustrating the identification of the origin, the tracing or the forfeiture of assets which he knows or must assume originate from a felony or aggravated tax misdemeanor shall be liable to a custodial sentence not exceeding three years or to a monetary penalty” (Swiss Criminal Code [SCC], 1937, art. 305 para. 1). Money laundering has three steps known as placement, layering, and integration (Lessambo, 2023). In the subsequent sections, these three stages and the term AML will be explained.

7.1.1 1st Step – placement

The initial phase of the money laundering process is placement (Lessambo, 2023). During this stage, illicit money is introduced into the financial system to eliminate direct association with illegal businesses (Lessambo, 2023). Large amounts of cash can be risky to hold onto, so the cash is typically placed somewhere safe such as a bank account (Lessambo, 2023).

7.1.2 2nd Step - layering

The second step of the money laundering process is layering, as Lessambo (2023) outlines. The goal of layering is to dissociate the money from its original illegal source by executing a series of complex transactions, which may involve electronic transactions, paper transactions, or even manual movement of the money (Lessambo, 2023). This step often involves transferring the funds across multiple countries, creating layers of transactions, and resulting in difficulties for prosecutors to trace the money back to its illegal origin (Lessambo, 2023).

7.1.3 3rd Step - integration

The last step of the money laundering process is called integration, according to Lessambo (2023). This final step involves returning the money to the criminal individual or organization in a way that appears legitimate from the outside (Lessambo, 2023). For example, a criminal may receive legal funds from the revenues of a business, such as a restaurant or a laundromat, that were acquired with illegal money (Lessambo, 2023). Another possibility of money integration may be the use of blockchain technology, where cryptocurrencies could be purchased with illegal funds.

7.1.4 Anti-money laundering

To gain a good understanding of the fight against money laundering and its processes, it is important to define the term “Anti-Money Laundering”. “AML is a set of directives [and laws] meant to deter, reduce and detect financial crimes within financial institutions and other organizations” (Wheeler, 2022, para. 6). In combating criminal activities, AML-related regulations have a crucial role in preventing funding of terrorists, the criminal use of identities, and transactions of illegal substances (Wheeler, 2022). The relevant sources for AML regulations and laws are the following:

- FATF recommendations;
- Basel Committee on Banking Supervision (BCBS) guidelines;
- Anti-Money Laundering Act (AMLA) of the Swiss Government & FINMA;
- Agreement on Swiss banks' Code of Conduct regarding the Exercise of Due Diligence (CDB 20) of the Swiss Bankers Association (SBA).

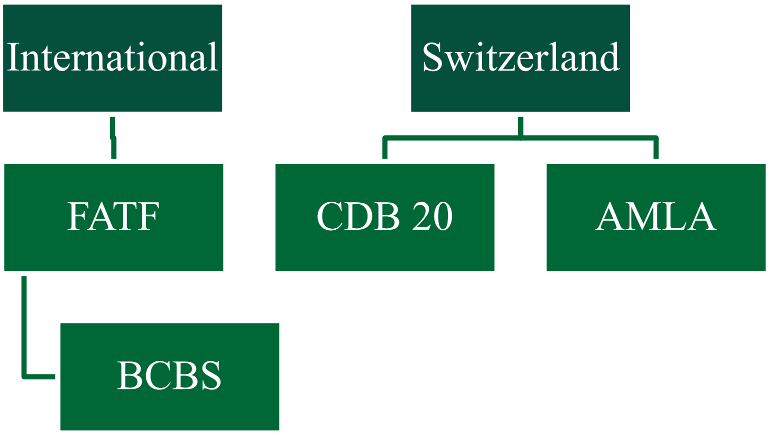
The following sub-chapter expands upon these regulations and laws to provide a comprehension.

7.2 Relevant regulatory frameworks and laws for AML and KYC

Financial companies in Switzerland, such as banks, have a broad customer base, catering to both national and international clients and providing their services domestically as well as abroad. Therefore, this section covers international guidelines such as the ones of the FATF and BCBS. Additionally, it examines the AMLA of Switzerland, as well as the code of conduct enforced by the SBA. These documents not only reference each other but also provide valuable insights into the AML topic. Figure 2 illustrates the connections between these regulations.

Figure 2

International and Switzerland's regulatory frameworks



Note. Own illustration.

7.2.1 Financial Action Task Force

FATF is an intergovernmental body whose standards encourage the enforcement of operational regulatory and legal measures to enhance the integrity of the international financial system (FATF, 2023). The FATF's international standard is applicable worldwide, however, its implementation may vary between counties due to differences in administrative and operational frameworks. The FATF (2023) recommendations outline several risk-related measures to detect and address money laundering, financing of terrorism, and proliferation of weapons of mass destruction. These recommendations include:

- policy development and domestic coordination;
- prosecution of policy development and domestic coordination;
- application of preventive measures in the financial and other specified sectors;
- definition of powers and responsibilities for competent authorities;
- enhancement of transparency and availability of beneficial ownership information;
- facilitation of international cooperation.

7.2.2 Basel Committee on Banking Supervision (BCBS)

The BCBS (2014) issued guidelines aimed at supporting the adoption of the FATF recommendations on a national level, where the guidelines combine the FATF's recommendations with the Basel core principles for banks and cross-border operations (BCBS, 2014). According to these guidelines, the main objective is to prevent and deter the use of banks for money laundering or terrorist financing, thereby protecting the reputation of banks and national banking systems. Additionally, these guidelines seek to uphold the integrity of the international financial system and support governments to combat corruption and terrorism financing.

7.2.3 Anti-money laundering act law in Switzerland

In the following, the different elements of the regulation in Switzerland are outlined. This includes the AMLA, the Anti-Money Laundering Ordinance (AMLO-FINMA), and the self-regulatory agreement CDB 20. The first part and the basis of the regulation is set by the federal act on combating money laundering and terrorist financing, also known as AMLA. This act builds the framework for the regulation. The second part of the regulation is defined by the ordinance of the

FINMA on combating money laundering and terrorist financing in the financial sector, commonly referred to as AMLO-FINMA. This ordinance focuses on certain parts of the AMLA but provides greater detail and specifies the regulation. The third component of the regulation is specified in art. 14 of AMLA (1997). It states that financial intermediaries need to build a self-regulatory organization (AMLA, 1997). In the case of banks, the self-regulatory organization is the agreement on CDB 20, issued by the SBA (CDB 20, 2020). For certain specific matters that remain unclear or need further clarification, FINMA also issues circulars (SBA, n.d.).

7.2.4 Swiss Bankers Association code of conduct

As previously mentioned, the CDB 20 holds significant importance within Switzerland. The Swiss regulation aims to implement the FATF recommendations through the CDB 20, which focuses mostly on areas not covered by the AMLA and the AMLO-FINMA (SBA, n.d.). In detail, the CDB 20 is a self-regulatory code of conduct that defines banks' duties to identify contracting parties and determine the identity of controlling persons or beneficial owners (CDB 20, 2020). On top of that, the CDB 20 actively assists in capital flight and tax evasion (SBA, 2023). In contrast to the international frameworks, violations of the CDB 20 can be punished with fines of up to CHF 10 million (SBA, 2023).

7.3 Know Your Customer in a regulatory context

KYC and CDD contribute significantly to achieving adherence to AML-compliant laws (Wheeler, 2022). According to a survey conducted by Thomson Reuters (2016), the continuous change of regulations and legislation poses a significant challenge for 87% of banks and 75% of investment managers. Consequently, measures and structures from AML regulatory frameworks that explain certain recommended practices and laws are closely related to KYC (Wheeler, 2022). KYC is typically considered part of AML since both processes are similar and occur during the onboarding stage (Wandelt & Werner, 2020).

The KYC process of a financial company is a costly undertaking (Thomson Reuters, 2016). According to a survey conducted by Thomson Reuters (2016), a financial company spends an average of $60 million to fulfill its obligations, but certain entities are allocating as much as $500 million to comply with KYC and CDD. This survey also indicates that the costs and complexity of KYC are on the rise, suggesting that the current numbers could be even higher.

Given the findings of this study and the crucial role of KYC and CDD, it is important to define the key terms. Wandelt and Werner (2020) define the KYC process as "checking the identity of your customer before [and while] doing business with them (i.e., onboarding the customer)" (p. 45). The KYC process involves identifying and verifying both new and existing customers based on AML requirements, as well as identifying the contracting party, determining (beneficial) ownership, and clarifying the economic background (Validatis, 2023). Additionally, the KYC process includes screening activities such as the screening of PEP, sanction screening, and AML screening (Wandelt & Werner, 2020).

Further, KYC consists of a CIP, CDD, and ongoing monitoring (Wheeler, 2022). To provide a more comprehensive explanation of CDD, the next section focuses on these terms, as well as risks and a risk-based approach, along with KYC storage solutions.

7.3.1 Customer identification program

Sullivan (2015) states that professionals in the financial industry typically use either KYC or CIP to refer to the entire process, leading to the terms often being used interchangeably. However, Sullivan (2015) makes a minor distinction between the terms: CIP enables financial institutions to acquire knowledge and comprehension about the parties involved in identity verification and business transactions *at the beginning* of the business relationship, whereas KYC assists financial institutions in gaining an understanding of the customer relationship and their financial behavior *during the entirety* of the business relationship.

KYC starts with the onboarding of a customer through a CIP to verify and identify the information of a customer (Wheeler, 2022). The FATF (2023) recommendations point out that financial institutions engaging in the process of customer identification and verification should utilize reliable and independent sources of information. The CIP-related pieces of information include, as explained among others in the account opening process (see table 1 and table 2), their full name, phone number, and ID documents (Wheeler, 2022). In addition, when the onboarding is carried out remotely, Sullivan (2015) suggests requesting additional documents, such as primary and secondary documents, conducting a counter-checked credit report, and making a phone call. Further responsibilities in a CIP are the maintenance of records, the description of information types, the consultation of government lists about terrorism, and implementing processes for these responsibilities (Sullivan, 2015).

For legal persons and arrangements, financial institutions should have a comprehensive understanding of the ownership and control structure of the customer (FATF, 2023). The AMLA (1997) emphasizes that the beneficial owner of a business relationship must be determined. This is reinforced by CDB 20 (2020), which describes that the controlling persons should be determined for a legal entity. CDB 20 further states that controlling parties can be identified by looking at voting rights, capital shares, or other discernible means. If the responsible party cannot be identified, the CDB 20 proposes that the chief executive manager is appointed as the party in charge. However, there are some exceptions: companies that are listed on a stock exchange, public authorities, as well as other banks are exempt from providing a declaration (CDB 20, 2020). If there is any doubt regarding the identity of the client or the beneficial owner, the steps must be repeated (AMLA, 1997).

7.3.2 Risks and a risk-based approach

According to Validatis (2023), the purpose of KYC-related investigations is to uncover unexpected risks in potential or current business relationships, such as bogus companies or funds from questionable sources. Therefore, we suppose it is important to consider various risks to gain insights into possible sources of criminal activities. To efficiently manage these risks, the regulatory frameworks recommend implementing a risk-based approach (FATF, 2023; BCBS, 2014).

An investigator working within a financial institution may encounter various types of risks that could be considered when creating a risk-rating matrix. Sullivan (2015) explains that during the initial stages of the KYC process, a comprehensive financial profile is established, encompassing individual, business, geography, and product risk. According to Sullivan, the individual risk involves factors such as the client's nationality, way of account opening, wealth, profession, and work location, in combination with the potential risk that may arise. Business risk, as Sullivan states (2015), pertains to the nature of the customer's business. He further describes that such risk could be associated with the money laundering ability of the business, its source of funding, as well as the utilized banking services of the client, and the overall plausibility of the client's business activities. The geographical risk, according to Sullivan (2015), relates to the origin of the money and where it goes, while product risk involves merchants importing and exporting goods from high-risk countries. The different risks associated with a customer are then assessed to determine a risk rating, which is further assembled into a risk-rating matrix (Sullivan, 2015). To our understanding, the entire process, from individual risks up to a risk-rating matrix, could be seen as a method of a risk-based approach because each risk must be evaluated depending on the impact of the overall risk profile of a customer.

The FATF (2023) recommendations propose to initially identify, assess, and understand the related risks for the financial system by adopting a risk-based approach. Further, the recommendations from FATF emphasize that a risk-based approach serves as an efficient prevention or mitigation of money laundering and terrorist financing. It is also suggested by the recommendations that countries should tailor their measures according to the level of risk, allowing for potentially less stringent measures in specific circumstances.

The BCBS (2014) guidelines extend the above-mentioned recommendations with additional bank-related information regarding the application of risk factors. BCBS (2014) lists that they are relevant at different levels, including country, sector, bank, and business relationship levels. However, it also highlights the importance of considering other levels for an extensive assessment in identifying the risk profile of a bank. These risk factors should be included in the policies and CDD process when the bank accepts, identifies, or monitors the customers throughout their relationship (BCBS, 2014). Additionally, the guidelines by BCBS (2014) explain the three lines of defense, which consists of the business units, the chief officer in charge, and the internal audit function. Consequently, these risk factors should not only be considered in the KYC process, instead, they should be applied to the entire bank. Further, the financial intermediary must implement organizational measures for adequate prevention of money laundering (AMLA, 1997). These measures include internal instructions for combating money laundering, regular training of the necessary employees, and establishing an internal AML body. The internal AML body assists the managers and line managers with the implementation of the regulation, prepares the internal instructions, and supervises the training (AMLO-FINMA, 2015).

7.3.3 Customer due diligence

According to Wheeler (2021), CDD is essential in AML and KYC undertakings, as it assists financial companies in preventing various financial crimes. CDD is defined as “the act of performing background checks and other screening on the customer to ensure that they are properly risk-assessed before [and after] being onboarded” (Wheeler, 2021, para. 2). Sullivan (2015) points out that investigators should perform sufficient due diligence to feel confident in their decisions and be capable of explaining them to the regulators. Furthermore, he recommends using a checklist during the CDD process to assist with the documentation. This is important because the investigators' work must be properly documented, as this can be crucial for the supervisory authorities in the event of an incident requiring evidence (Sullivan, 2015). The frequency of conducting reviews is based on the level of risk associated with the accounts (Sullivan, 2015):

- high-risk accounts should be reviewed annually;
- medium-risk accounts should be reviewed every two to three years;
- low-risk accounts should be reviewed every three to five years.

In addition to scheduled reviews and after establishing a business relationship, CDD is recommended if there is a significant change of a client's situation or if a trigger event occurs (Sullivan, 2015).

There are four scenarios in the FATF (2023) recommendations in which CDD is mandatory. The first one is when a business relationship is established. The second scenario is if occasional transactions exceed the established threshold of 15'000 USD or when wire transfers are involved. The third one can be caused by suspicion of money laundering or terrorist financing. The fourth scenario is when the financial institution has doubts about the already gathered data regarding customer identification.

According to BCBS (2014) guidelines, the CDD addresses both bank customers and persons acting on their behalf. They also tell banks to establish a systematic procedure for customer identification and emphasize that a bank should not conduct business with a potential customer before completing the identity verification process. However, the BCBS (2014) guidelines also state that if CDD cannot be completed, it can be carried out later while ensuring suitable risk management practices to avoid disrupting regular business operations. In such cases, it further states that if difficulties arise in the verification of the customer's identity, the bank is recommended to close the account or block the access to it.

7.3.4 Different levels of due diligence

CDD can be divided into simplified due diligence, standard due diligence, and Enhanced Due Diligence (EDD) (Wheeler, 2022). According to Sullivan (2015), the risk matrix of a customer helps to determine the extent of due diligence. Further, he mentions that lower risk ratings result in a less extensive due diligence process, while medium or high ratings of risks may require a more comprehensive approach.

To gain a comprehensive grasp of the various tiers of due diligence, the different levels are described below:

- **Simplified due diligence** is one of the most executed levels of CDD and is usually appropriate when the likelihood of money laundering or terrorist financing is minimal or negligible (Chitimira & Munedzi, 2022). Therefore, simplified due diligence is carried out when a customer poses a minimal degree of risk (Wheeler, 2022). According to Wheeler (2021), the only obligation is to check the government ID, but identification does not need to be verified.
- **Standard due diligence** falls between simplified due diligence and enhanced due diligence. Therefore, standard due diligence is conducted when a customer poses a moderate level of risk (Wheeler, 2022). According to Wheeler (2021), verifying the identity of a customer should be done using an independent and trustworthy source.
- **Enhanced due diligence** involves a more thorough investigation into the background of an individual or company (Sullivan, 2015). Therefore, EDD is performed when customers pose a high level of risk (Wheeler, 2022). According to Wheeler (2021), additional information should be obtained regarding the customer's ID, source of funds, business relationship, and transaction reasons. She also states that monitoring should be performed on an ongoing basis. The EDD suggestions by Wheeler (2021) align with the BCBS (2014) guidelines, which describe that EDD should be conducted in cases of higher risk of money laundering or when there is a need for a deeper understanding of the client, transactions or source of funds.

There are additional actions aimed at PEPs and problematic jurisdictions provided by the BCBS (2014) guidelines and the AMLA (1997). The guidelines emphasize being cautious when dealing with jurisdictions that have known issues with AML and countering the financing of terrorism, and thus, enhanced due diligence should be applied. Regarding client databases, the BCBS (2014) guidelines recommend that banks should regularly check for PEPs and high-risk accounts. They further suggest conducting EDD or a review as soon as there are changes in sanction lists.

The AMLA (1997) provides an interesting list of higher-risk business relations that we assume to be relevant to determining a high-risk account. The reason is that the law states that higher-risk business relations are business relationships with PEPs and their social environment or business relationships with people that are residents in a country that has been marked as high-risk or uncooperative by the FATF (AMLO-FINMA, 2015).

The FATF (2023) recommendations point out additional measures for PEPs. According to the recommendations, the additional measures are as follows: One measure is having a suitable risk-management system so that the politically exposed client or the politically exposed beneficial owner can be determined. Another measure is an obstinance of approval from the senior management, then there should be an establishment of an appropriate measure to detect the source of wealth and funds. The last measure is about enhanced repeated monitoring.

7.3.5 Ongoing monitoring

As this paper showed already in the section about the payment process, it is important to cover transaction monitoring as a part of KYC. If an account bears a higher risk, financial companies should conduct ongoing monitoring of transactions, usually in real-time (Wheeler, 2022). To monitor customers, there is software to detect patterns and flag unusual transactions that are then analyzed by the AML business unit to assess the degree of suspicion (Sullivan, 2015). The flag of a transaction is a trigger event, according to Sullivan, and is checked for resolution or escalation for an additional review. This review might result in a suspicious activity record or termination of the business relationship (Sullivan, 2015).

In alignment with Wheeler (2022) and Sullivan (2015), the BCBS (2014) guidelines have some additional input. It states that it is important for the risk management of banks to understand casual banking-related activities reasonably to identify unregular transactions. According to BCBS (2014) guidelines, if the bank does not have this expertise, it may result in the inability to report dubious transactions. Furthermore, BCBS (2014) recommends implementing monitoring systems to detect them. To assess the inherent risk of different clients, collection of accounts, patterns of transactions, and application of products adequately, it is recommended to apply CDD and utilize law enforcement data (BCBS, 2014).

According to Alkhalili et al. (2021), transaction monitoring with machine learning could be helpful in fighting financial crimes. The authors describe that financial companies use AML software, sanction screening, and watch-list filters to prevent transactions with blocked accounts. To enhance this process, they suggest implementing machine learning to get better results in a shorter amount of time. Therefore, they introduce an architecture to build and integrate their machine-learning elements into a watch-list filter of an AML system. However, according to Alkhalili et al. (2021) the industry has concerns about the automation of compliance because offenses can be fined heavily. Despite that, based on Alkhalili et al (2021), the concerns may be diminished. Thus, machine learning could be important in the future of ongoing transaction monitoring.

7.3.6 KYC storage solutions

As previously shown, the KYC process involves gathering various information. According to Pratz et al. (2021), there are currently the following four storage approaches to designing AML and KYC processes for banks:

- **Bank-internal optimization:** In the first approach, the bank maintains its systems to manage the KYC process (Pratz et al., 2021). This approach ties up a lot of resources from the bank's point of view because the entire process is carried out by the bank itself.
- **Managed service:** In the second approach, the bank relies fully or partially on services from a third-party company (Pratz et al., 2021). Thus, the bank buys certain compliance services from a third-party provider (Pratz et al., 2021). This is beneficial because it ties up fewer of the bank's resources than the first approach. However, it is not suitable for reviewing multinational corporate customers since there is hardly any third-party provider that offers this compliance service across several jurisdictions in high quality (Pratz et al., 2021).
- **Utility service:** In the third approach, a group of banks receive standardized compliance services from a third-party provider (Pratz et al., 2021). Unlike the second approach, the economies of scale come into play here. This is particularly advantageous if the participating banks share many international customers. However, in practice, there is very rarely a substantial common customer base among the participating banks, which is why this approach is not as effective and efficient as in theory (Pratz et al., 2021). On top of that, there is often a lack of standardization among the participating banks, which is not conducive to this approach (Pratz et al., 2021).

- **Data-sharing network:** Contrary to the second and third approaches, there is not just a single data provider (Pratz et al., 2021). Hence, the banks are directly sharing their KYC data (Pratz et al., 2021). This makes this approach scalable. Despite its advantages, this approach has not yet established itself due to a lack of acceptance by the regulatory authorities and a lack of standardization of data and IT systems (Pratz et al, 2021).

A survey on AML conducted by Thomson Reuters (2022) shows that banks increasingly rely on third-party service providers, many of which provide automated solutions to streamline processes. However, these automated solutions are not yet well established, which is why there are still many unresolved issues in this regard (Thomson Reuters, 2022).

7.4 AML Prosecution

After explaining AML and KYC within a regulatory context, it is important to explain in this subchapter what happens if the bank has, based on the gathered information, reasonable grounds to suspect that a business relationship is used for money laundering. This section starts by explaining the definition of reasonable grounds, afterwards explaining the reporting process, and lastly, the prohibition of information and its exceptions.

As the term “reasonable grounds” to suspect money laundering is broad, this paragraph is about how this is defined in the case of money laundering. Reasonable grounds to suspect means that there is either specific evidence or there are several indications that the business relationship is being used for money laundering, and the suspicion could not be rooted out through additional clarifications (AMLA, 1997). If that is the case, the bank is obliged to file a report with the money laundering reporting office in Switzerland (referred to as the reporting office from now on) according to the AMLA (1997). This also applies if the bank terminates the process of establishing a relationship because of suspicion of money laundering (AMLA, 1997).

The FATF (2023) recommendations state the importance of the fact that the institutions and the personnel do not face criminal and civil penalties for violating restrictions on disclosure by contract, law, and regulation. Additionally, according to FATF (2023), this should also be applied to cases in which the exact criminal background is unknown or the criminal activity effectively occurred. This is supported by the law in Switzerland as art. 11 of the AMLA states: “Any person who in good faith files a report under art. 9 of this act or who freezes assets by art. 10 may not be prosecuted for a breach of official, profession or trade secrecy or be held liable for breach of contract." (AMLA, 1997, art. 11).

While the reporting office analyses the case, the bank continues to execute customer orders (AMLA, 1997). The only condition is that orders regarding significant assets are only executed in a way that allows prosecutors to follow the trail of the assets (AMLA, 1997).

After filing a report with the reporting office, the reporting office can transmit the case to a prosecution authority (AMLA, 1997). If the transmission takes place, the bank must immediately freeze the related assets (AMLA, 1997). The assets are frozen until the prosecution authority sends an order or a ruling to the bank. This can last up to a maximum of five working days after the bank was informed of the transmission (AMLA, 1997). If the bank is not informed by the reporting office that the case has been given over to a prosecution authority within 40 days, the bank is allowed to terminate the business relationship (AMLA, 1997). If the bank terminates the business relationship, it must be reported to the reporting office, and the asset withdrawal has to be traceable (AMLA, 1997). This concurs with the BCBS (2014) guidelines which state that banks should have the ability to freeze assets of specific subjects in alignment with national law and the United Nations security council resolutions.

It is prohibited to inform the customer or third parties that a report has been filed (AMLA, 1997). Exceptions to this are authorities and organizations responsible for supervision, people carrying out audits as part of the supervision, and other financial intermediaries if it is required for the freezing of the assets or when providing joint services (AMLA, 1997).

7.5 Virtual assets

In the first four sub-chapters, we covered the AML and KYC system in a traditional monetary system. Since we aim to create a concept and smart contract to regulate the crypto market, we initially want to present quantitative data about cryptocurrency that is laundered. Then, we provide relevant information from a regulatory and legal perspective in the context of virtual assets. Ultimately, we finish the chapter by describing the problem that our KYCrypto concept and smart contract may solve.

According to a report on crypto crime by Chainalysis (2022), the total cryptocurrency value laundered amounts to $8.6 billion in 2021. Importantly, this figure only considers cybercriminal activities such as ransomware attacks. Hence, illicit transactions from criminal activities, such as drug trafficking or human trafficking, are not even included here because they cannot be quantified, according to Chainalysis (2022). This clearly shows that there is a need for action here. Therefore, in the following, we analyze the current situation concerning AML in the crypto market.

In recent years, the FATF has further clarified its recommendations regarding the regulatory treatment of virtual assets and adapted them to ongoing technological progress (FATF, 2023). In doing so, the FATF (2023) also issued new recommendations about so-called Virtual Asset Service Providers (VASP), such as crypto exchanges or wallet providers. According to them, these VASPs should now also ensure that AML and KYC policies are applied to their customers, just like traditional financial institutions must do.

The abovementioned FATF recommendations have been recently implemented or are about to be implemented by some jurisdictions (Mulhim, 2022), one of which is Switzerland (Landtwing-Leupi, 2020). According to Landtwing-Leupi (2020), the abovementioned FATF recommendations about VASP are also implemented in Swiss legislation (AMLA, 1997). Based thereon, the same obligations exist for VASP as for traditional banks. Furthermore, payment tokens such as ether are regarded as virtual assets, and therefore they are also subject to AML measures according to AMLA (Landtwing-Leupi, 2020).

Even though some jurisdictions have already fully implemented the abovementioned FATF recommendations, there are still ways to buy cryptocurrencies without going through KYC verification (PXL Vision, n.d.). And even in jurisdictions that have already implemented them, many financial institutions (e.g., banks, asset managers, crypto exchanges, etc.) are still struggling to apply AML and CDD rules to the crypto market, according to a survey conducted by Thomson Reuters (2022).

To conclude, in recent years, the crypto market has become increasingly regulated as shown above. However, according to our understanding, the lack of implementation, the global differences in the intensity of the fight against money laundering, as well as the fundamental characteristics of blockchain technology, such as pseudonymity, result in the crypto market not yet being as comprehensively regulated as the conventional financial market.

8. Literature review about blockchain-based AML and KYC

After having described the traditional AML and KYC regulations in the previous chapters, in this literature review, we provide an overview of the related blockchain technology topics to identify the current research gap. The starting point of our literature review is the paper of Malhotra et al. (2022), as they already provided a PRISMA-guided systematic review of the topics we want to cover in this chapter. The PRISMA statement is a research method to support literature reviews in a standardized, transparent, and complete manner (Page et al., 2021). Additionally, we found other sources to complement this systematic review which is from another databank or got more recently published.

This literature research chapter is divided into five sub-chapters: Blockchain-based KYC storage, on-chain and off-chain based KYC frameworks, blockchain-based AML, privacy-preserving KYC exchange, and the research gap.

8.1 Blockchain-based KYC storage

In the ethereum storage-based frameworks section of the literature review of Malhotra et al. (2022), there are 18 papers describing smart contract implementations. In this literature review, two papers propose storing KYC-related data off-chain because of a limiting on-chain storage factor, and three papers suggest storing the data encrypted as a javascript object notation format in the InterPlanetary File System (IPFS). Since there are a significant number of authors discussing the topic of data storage, we want to investigate what these storage solutions are.

Parra-Moyano and Ross (2017) present an approach to facilitating KYC-related data between banks and regulating authorities. In the initial step of their approach, a customer gives the desired bank the relevant KYC documents, and the bank stores them in its local database. Afterwards, they describe that the document is transformed into a hash, and this hash is then uploaded into a distributed ledger. As a last step, the customer can give access to other institutions via their specific smart contract (Parra-Moyano & Ross, 2017). In their approach, we appreciate the fact that a customer has to some extent control in deciding whether to provide their information to another bank or not. Understanding it from a KYC investigator's point of view that probably slows down the process. However, from a data privacy point of view, there are potentially fewer difficulties involved.

To our understanding in the paper of Parra-Moyano and Ross (2017), the regulating authority acts as a central role and provides a private blockchain. Therefore, their first approach seems to be a *centralized file storage* approach. The authors criticize in their first approach that the regulating authority could be a victim of a cyberattack, insider fraud, or corruption. To refute their criticism, they argue to decentralize the first approach and create a second approach by cutting the regulator. However, they mention in their exchange with experts that the benefit of financial stability with regulating authorities may outweigh their criticized risk.

Singhal et al. (2020) present in their paper an approach using the IPFS technology, which is a *decentralized file storage* approach. Their paper describes the second approach of Parra-Moyano and Ross (2017) in more detail, and it seems to be on a more recent research basis. Singhal et al. (2020) explain that a file can be uploaded on the IPFS node, which sends back a hash that points to the file in the node. Further, they describe the hash should be stored on the blockchain to be then verified. The advantage, according to the authors, is that the governance is enhanced because changes can be tracked. Accessibility is better as well, and the method to save the hash on the blockchain makes this system efficient (Singhal et al., 2020). However, according to the feedback received from the finance executives in the paper of Parra-Moyano and Ross (2017), banks seem not to want the private information of a customer in the form of a hash on a publicly distributed ledger because mistakes in code could lead to reverse engineering and private information could be published. Despite that, a counterargument could be that the paper of Singhal et al. (2020) is more current, and we assume that the technology has evolved. Due to this reason, the argument of Parra-Moyano and Ross (2017) could lose its validity over time.

In our opinion, there might be legal and technical challenges regarding the sharing of sensitive AML and KYC-related data. Without these challenges, different banks could get the data needed for their KYC process by using a blockchain storage system with the permission of the customers. Therefore, not every bank has to gather the data separately, and overall costs could be reduced (Parra-Moyano & Ross, 2017). As a result, we acknowledge the previous work regarding a blockchain-based KYC storage solution that may have the potential for a paradigm shift.

8.2 On-chain and off-chain based KYC frameworks

An alternative approach discussed in the paper by Yadav and Chandak (2019) is to involve a combination of on-chain and off-chain components in the KYC process. The authors propose a solution that involves creating a blockchain-based system consisting of a client-based mobile application and a bank-based website. When the user installs the application and signs up, providing their personal information, the whole proposed process by Yadav and Chandak (2019) starts. Once the user is registered, they need to fill out the KYC form, including several personal information, such as their name, phone number, permanent account number, and a unique identification number, named Aadhar, a number which in this paper is issued by the Indian government (Yadav & Chandak, 2019). After filling out this personal information, the data is temporarily stored on an Amazon web services server and automatically verified using the permanent account number and Aadhar (Yadav & Chandak, 2019). The authors suggest that after the authentication of the customer's details, the data is added to the blockchain using the ethereum application programming interface and a smart contract written in solidity language. The customer will be notified after the verification process and then can choose a bank where they want to open an account, and their details are encrypted using the bank's public key and sent to the chosen bank for review (Yadav & Chandak, 2019).

The approach by Yadav and Chandak (2019) combines the on-chain and off-chain components in the KYC process and has valuable implications for the AML and KYC blockchain literature, particularly with their figures that illustrate the implementation of the KYC process by using smart contracts. These illustrations benefit the overall understanding of customer onboarding in a blockchain-related CIP.

8.3 Blockchain-based anti-money laundering

Several authors are exploring the potential of blockchain technology in the field of AML. Oad et al. (2021) present a blockchain-enabled transaction scanning method to detect anomalous activities in financial transactions while using clearly stated assumptions. These are needed to detect so-called outliers, defined as data points that behave differently than within the spread of the assumed rules (Oad et al., 2021). The contrary is an inlier, which is a non-concerning data point (Oad et al., 2021). Defining outliers is challenging due to their high diversity and unpredictability, however, inliers tend to be stable and are, therefore, valuable for defining outliers (Oad et al., 2021). The accuracy of the approach from the paper from Oad et al. (2021) is validated through the execution of an algorithm using a mock transaction history. The results show that the outlier detection method performs better than rapid movement funds (Oad et al., 2021). The paper gives us valuable information, however, one concern that we have is in the evaluation of the method because it relies only on simulation results with a mock transaction history. To our understanding, it could be the case that real-world transactions are much more complex, and as a result, the method would not fully represent the current world.

A similar blockchain-based approach to detecting anomalies is described in the paper of Xu et al. (2021). The authors apply a research design to create a blockchain-based AML system for the CDD process. In their system, secure storage and sharing of CDD information are provided within an accepted blockchain network, meaning that most participants, who have approval, have access to the data of the customers. Additionally, the system ensures real-time updates of customer data and uses smart contracts to automate AML processes and improves the efficiency of AML audits (Xu et al., 2021). Further, they describe that if any suspicious activity occurs in the future, smart contracts have the function to alert and trigger notifications to AML auditors (Xu et al., 2021). Therefore, their implementation contributes significantly to the existing literature by showing that smart contracts play a vital role in a blockchain-based AML system. It also addresses the implications for regulators, as the system enables ongoing surveillance of suspicious activities. However, the implementation regarding the coordination of information in a standardized format could be conflicting among different financial institutions, and legal issues can arise related to data protection (Xu et al., 2021).

Visualization is another way to detect suspicious activity, according to Singh and Best (2019). This study showcases the application of link analysis in the detection of suspicious bank transactions, highlighting its effectiveness in identifying potential instances of money laundering. To validate this approach, a prototype application (AML^{2}ink) is employed and used (Singh & Best, 2019).

The paper from Singh and Best (2019) makes a valuable contribution by exploring the use of data visualization techniques for detecting money laundering. However, the approach relies on access to the bank transaction data of the entity under investigation (Singh & Best, 2019). This means it is possible to detect the risk within the institution itself but much harder outside of the entity due to the data concerns (Singh & Best, 2019). In contrast, a bank investigator can analyze and report suspicious activities not only within an institution but also between them since this person has no restrictions regarding data concerns (Singh & Best, 2019). Not being able to track the money and only being able to identify money laundering within the company is a good approach, but one that is not sufficient in today's complex world, as customers nowadays usually have more than one bank account, and these usually with more than one bank (Singh & Best, 2019).

8.4 Privacy-preserving KYC exchange

The paper of Biryukov et al. (2018) introduces in their Know Your Customer Exchange (KYCE) an interesting concept of using a whitelist. Further, they use a method to cryptographically ensure the privacy of whitelisted addresses by using a cryptographic accumulator. Their description of an accumulator is based on the construction of Camenisch et al. (2009) and is, as far as we know, a way to mathematically ensure anonymity. In the paper of Biryukov et al. (2018), they argue about a problem regarding the whitelist due to its visibility on the public blockchain. Furthermore, it is problematic for them that a third party can analyze the transactions of the whitelisted addresses.

In the paper of Biryukov et al. (2018), two use cases are described, and the existence of a proof-of-concept is briefly mentioned. Both use cases are according to the paper about trading tokens on an exchange. The difference in their two use cases is that; in one the exchange is KYC-compliant and in the other the token is KYC-compliant. The proof-of-concept implementation was created by the winning team of the Luxblock hackathon called CryptoLUX (Biryukov et al., 2018). By analyzing the proof-of-concept and in the context of the paper of Biryukov et al. (2018), we conclude that their approach is about trading different assets by using a cryptographic whitelist and a smart contract that acts like an exchange.

The privacy aspect of the paper of Biryukov et al. (2018) is well explained, and they include several KYC elements, but it seems to exist a scalability and governance issue. In their paper, they suggest that one entity is in control of the onboarding. Further with the knowledge, based on the paper and proof-of-concept, we deduce that it may be the same case that only one entity is involved in the monitoring of the transactions, probably due to the privacy aspect. Based on these findings, we conclude that the CIP and monitoring elements of KYC are included in their paper. Biryukov et al. (2018) even include a control process so that certain customers cannot use their tokens. However, in the traditional system, several entities are involved in conducting CIP, CDD, monitoring, and prosecution. To our understanding, the privacy features of the mathematical solution may interrupt the KYC process and vice versa by focusing on only one conducting entity. Therefore, we assume that KYCE may have a disadvantage in scalability and governance because the responsibilities may not be dividable.

A general topic to be discussed is the severance of privacy concerns in the paper of Biryukov et al. (2018). Overall, we pose the question if the inherent blockchain technology aspects are not already anonym enough for a customer. However, what they contribute, and we acknowledge is the anonymized process of whether a customer is KYC compliant or not, the ability that a user receives a unified identity, and the circumstance that only one KYC provider has the information about a customer. Additionally, the information of a customer is, according to the paper by Biryukov et al. (2018), not uploaded into the blockchain. This adheres to the paper of Parra-Moyano and Ross (2017) regarding the argument for not saving customer documents on the blockchain. In conclusion, Biryukov et al. (2018) contributed a valid concept and an interesting proof-of-concept in their smart contract to the AML and KYC blockchain literature.

8.5 Research gap

In the previous four sub-chapters, we analyzed and critically discussed certain papers of the PRISMA-guided systematic literature review of Malhotra et al. (2022). Each paper has its important contribution to the overall AML and KYC blockchain literature, but we did not find a paper that discusses the following important aspects of a holistic concept:

- An approach that may not have certain data privacy issues and is more focused on the overall KYC measures instead of storage solutions;
- An illustrative approach in explaining different cases in the KYC process with on-chain and off-chain processes;
- An approach that aims to use already existing structures for KYC processes like transaction monitoring;
- A concept that covers CIP, CDD, monitoring, and AML prosecution in a blockchain environment by using a smart contract and an oracle.

Of all the analyzed papers the one by Biryukov et al. (2018), about the privacy-preserving KYC exchange, covered the most of the above-mentioned aspects. However, the intended nature of their approach is to provide a crypto exchange that we do not aim to recreate. Therefore, we identified the research gap in the above-mentioned aspects by conducting KYC in a regulatory environment that aims to be AML compliant. We think closing this research gap would be a chance for a possible paradigm shift in the combat against money laundering.

9. KYCrypto concept – implementation

We have already shown in detail how the legacy system is designed in terms of account opening, account management, transaction, and the topics of AML and KYC within regulatory frameworks and laws. We have also analyzed the various use cases and approaches regarding the partial or complete transfer of legacy systems to a blockchain-based solution by reviewing the existing literature. In the following, we present in detail our own solution and approach for a partial transfer of the analyzed systems to a blockchain-based solution. Firstly, we show the basic principles of our concept before introducing our newly created business model. Following that, we present our onboarding process and outline how customers can purchase or transfer our self-developed contract token. Lastly, we explain how we make sure that our concept is compliant with AML and KYC policies.

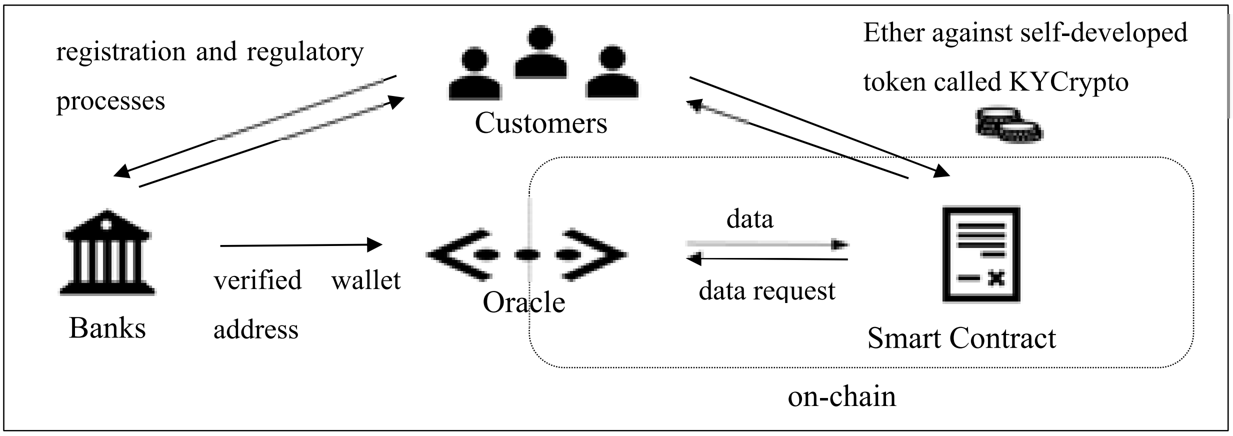
On the one hand, the goal of our solution is to increase the efficiency of the legacy processes by applying the advantageous features of blockchain and, on the other hand, to create a new business model for banks.

9.1 Basic principles

In the following, we show the basic principles of our solution. The detailed steps are shown in the steps below.

Figure 3

Basic principles


registration and regulatory processes
Customers
Ether against self-developed token called KYCrypto
Banks
verified wallet address
Oracle
data
data request
Smart Contract
on-chain

Note. Own illustration.

As mentioned above, our aim is to combine the existing procedures and processes with the advantages of blockchain technology. One of the goals is to create a cryptocurrency compliant with the same regulations and processes against money laundering as traditional currencies. For this purpose, we created our own token based on a self-developed smart contract. We named our token “KYCrypto.” The naming of our token comes after KYCrypto refers to the regulatory policy of KYC and the abbreviation of cryptocurrency.

Our smart contract runs on the ethereum virtual machine. Therefore, we developed our token according to the Ethereum Request for Comment 20 (ERC-20) standard. The purpose of this newly created token is to represent a currency that is directly linked to the value of ether. The value of a unit of the newly created currency is thus linked to ether with a fixed exchange rate of 1:1.

Our solution ensures that the KYCrypto complies with the AML and KYC-related regulatory processes by linking the information from the banks to our smart contract via oracle. The purpose of the oracle is to ensure that the smart contract has access to the banks' data. The banks, in turn, operate their own control processes off-chain to be compliant with the KYC-related processes and the AML laws, whereby they make their results available to the smart contract via oracle.

Finally, our smart contract ensures that only people checked by the banks can purchase, transfer, or receive KYCrypto. Another advantage that our solution offers is that it will be easier for banks to monitor people's transactions in the future. Because unlike the traditional system, all blockchain-based transactions can be monitored publicly, whereas in the traditional system, cash-based transactions, for example, cannot be traced.

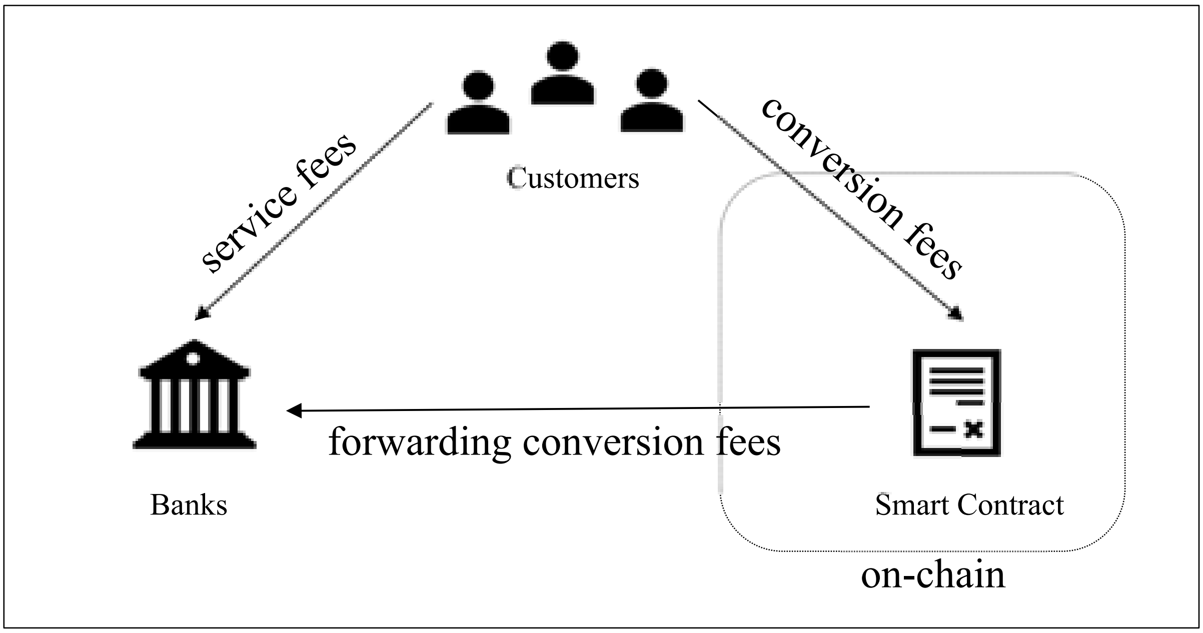
9.2 Newly created business model for banks

As shown in chapter 7.3, one of the challenges the banking industry is facing lies in the increasing compliance costs. The costs incurred by the AML and KYC amount up to 600 million euros for Swiss banks, according to a study by Pratz et al. (2021). As shown in the study by Pratz et al. (2021), not only are costs incurred by AML and KYC checks ever increasing, but also the number of fines to be paid due to not complying with these regulations. On top of that, this study shows that bank customers are not satisfied with the way banks manage their regulatory processes regarding AML and KYC because there are too many redundant steps from their perspective. To sum it up, ever-increasing costs, the risk of being fined, and unsatisfied customers are the pressing matters in this relation.

Our concept presents a practical and functional solution to the increasing compliance costs by introducing new revenue streams, as illustrated in the figure below.

Figure 4

Business model



Note. Own illustration.

To date, banks have not generated any revenues from services rendered in the context of AML and KYC checks. In our solution, the participating banks charge service fees for conducting the CIP, CDD, for the verification of the customer's wallet address during the onboarding process, and for ongoing monitoring. The service fees are settled off-chain. Hence, the customers apply to pay them directly to the banks that carry out the onboarding process. The level of fees is determined independently by each bank which is why its level can vary from bank to bank. Furthermore, this service fee represents a one-time fee.

In our concept, we introduced another fee. This fee is settled on-chain via smart contract. Every time a customer buys or sells KYCrypto in exchange for ether a fee of 1% on the purchase and sales value will be levied. Our smart contract collects this conversion fee directly on every purchase and sale of KYCrypto. Subsequently, the smart contract forwards these conversion fees to the bank using a distribution key. The distribution key is based on the number of AML and KYC checks performed by banks. The checks for international corporate customers are weighted higher because they are more time-consuming, according to the study Pratz et al. (2021).

Since our solution introduces two revenue streams related to the compliance services rendered by banks, we mitigate the problem of ever-increasing compliance costs.

9.3 Onboarding and account opening

In this section, we show the way customers are onboarded within our KYCrypto concept. The onboarding is one of the most important stages when it comes to implementing AML and KYC policies. Additionally, we illustrate our solution with an activities diagram.

9.3.1 General

In the following, we refer to customers as users or potential users of our solution. The onboarding process differs between customers who already have an account with a participating bank and customers who do not yet have such an account. Based thereon, we illustrated the two distinct onboarding processes in the figures below.

Once a person has decided to participate in our solution, i.e., to purchase KYCrypto, they must go through an appropriate onboarding process. In both cases, i.e., regardless of whether the customer already has a bank account or not, the first step is to approach a participating bank. However, the specific approach differs here, depending on whether the customer already has a bank account or not. We will go into these specific peculiarities in the sections below.

As a matter of principle, onboarding a customer who does not have a bank account yet is more time-consuming and comprehensive than for a customer who already has an existing bank account. However, as can be seen in the figures below, the bank performs CDD in both cases (see section 9.7.1). Consequently, in case of suspicious activities, the bank reports this to the competent authorities as already explained in chapter 7.4.

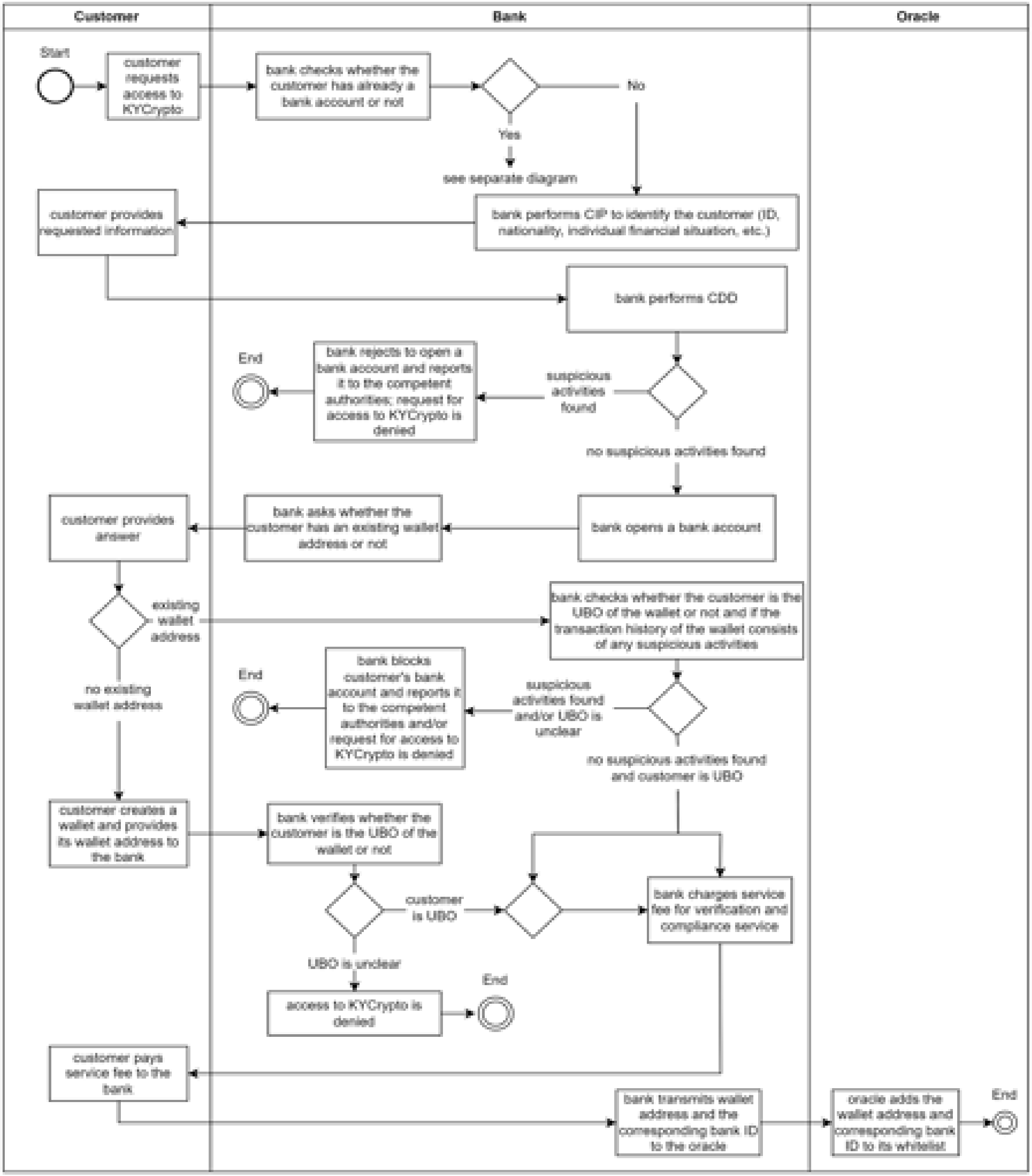
Another common feature of the two onboarding processes is that in both cases, the customer must provide the bank with a wallet address. Here we refer to our explanations in section 9.7.1. If the customer does not yet have a wallet address, he must first create a wallet. The bank also applies the CDD principles to the wallet (see section 9.7.1 for further details). If no suspicious activities are found and the beneficial owner also corresponds to the customer, the bank forwards the verified wallet address and the corresponding bank ID to the oracle, subject to the payment of the service fee from the customer to the bank. The final step the oracle enters the received information automatically into a register. In the following, we call this register "whitelist." This means that all customers who have successfully completed the onboarding process are whitelisted.

9.3.2 Onboarding without an existing bank account

In this section, we only mention the specific peculiarities related to the onboarding process of customers without an existing bank account. The detailed onboarding process is outlined below in the figure 5. One peculiarity of this process is that the customer must first open a conventional bank account. Another peculiarity forms the fact that the bank performs a CIP as part of the account opening.

Figure 5

Onboarding of a customer without an existing bank account



Customer
Bank
Oracle
Start
customer requests access to KYCrypto
bank checks whether the customer has already a bank account or not
No
Yes
see separate diagram
customer provides requested information
bank performs CIP to identify the customer (ID, nationality, individual financial situation, etc.)
bank performs CDD
End
bank rejects to open a bank account and reports it to the competent authorities; request for access to KYCrypto is denied
suspicious activities found
no suspicious activities found
customer provides answer
bank asks whether the customer has an existing wallet address or not
bank opens a bank account
existing wallet address
bank checks whether the customer is the UBO of the wallet or not and if the transaction history of the wallet consists of any suspicious activities
no existing wallet address
End
bank blocks customer's bank account and reports it to the competent authorities and/or request for access to KYCrypto is denied
suspicious activities found and/or UBO is unclear
no suspicious activities found and customer is UBO
customer creates a wallet and provides its wallet address to the bank
bank verifies whether the customer is the UBO of the wallet or not
customer is UBO
bank charges service fee for verification and compliance service
UBO is unclear
End
access to KYCrypto is denied
customer pays service fee to the bank
bank transmits wallet address and the corresponding bank ID to the oracle
oracle adds the wallet address and corresponding bank ID to its whitelist
End

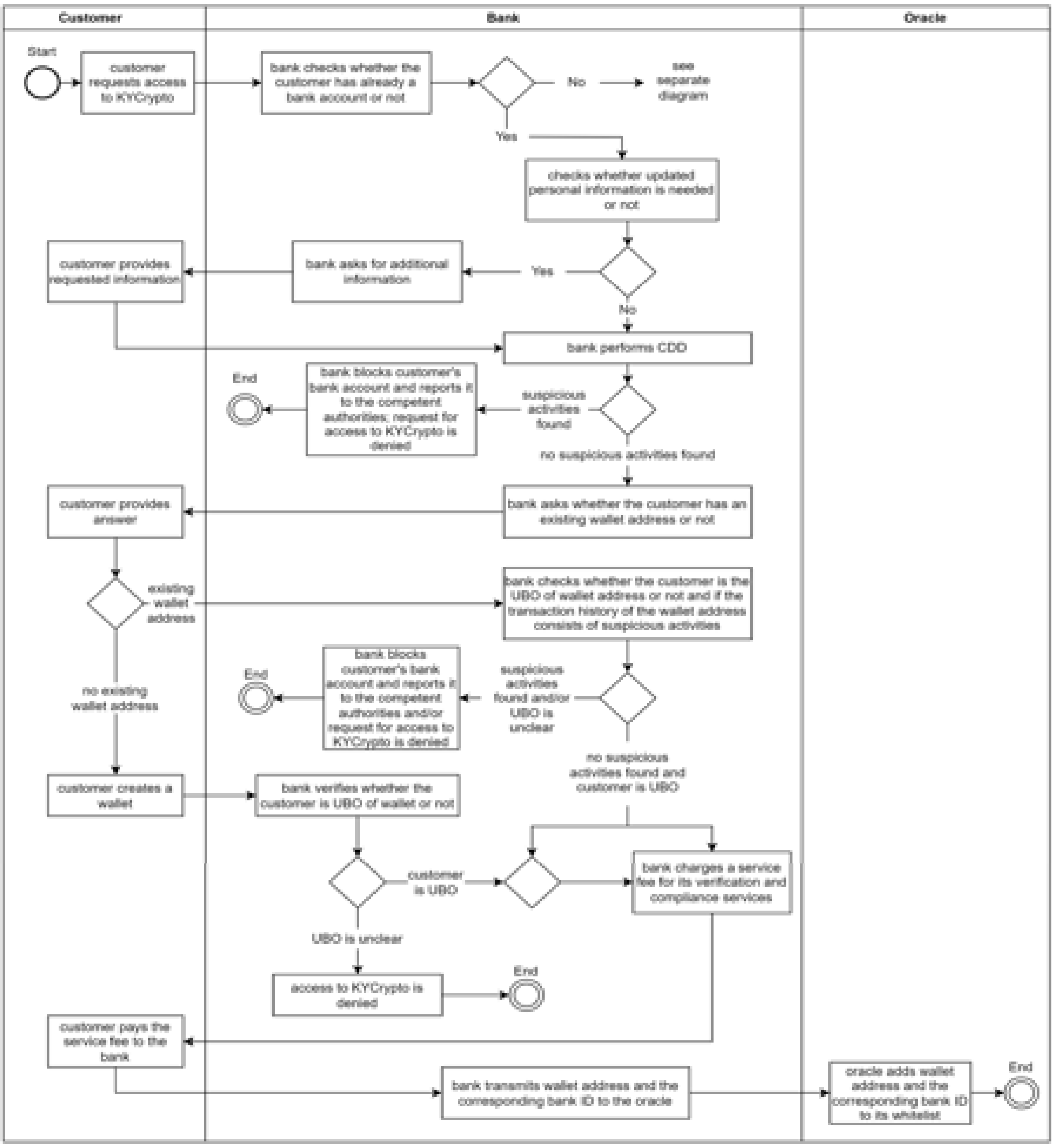
Note. Own illustration.

9.3.3 Onboarding with an existing bank account

The detailed onboarding process of a customer who already has a bank account is shown below.

Figure 6

Onboarding of a customer who has an existing bank account


Customer
Bank
Oracle
Start
customer requests access to KYCrypto
bank checks whether the customer has already a bank account or not
No
see separate diagram
Yes
checks whether updated personal information is needed or not
customer provides requested information
bank asks for additional information
Yes
No
bank performs CDD
End
bank blocks customer's bank account and reports it to the competent authorities; request for access to KYCrypto is denied
suspicious activities found
no suspicious activities found
customer provides answer
bank asks whether the customer has an existing wallet address or not
existing wallet address
bank checks whether the customer is the UBO of wallet address or not and if the transaction history of the wallet address consists of suspicious activities
End
bank blocks customer's bank account and reports it to the competent authorities and/or request for access to KYCrypto is denied
suspicious activities found and/or UBO is unclear
no existing wallet address
no suspicious activities found and customer is UBO
customer creates a wallet
bank verifies whether the customer is UBO of wallet or not
customer is UBO
bank charges a service fee for its verification and compliance services
UBO is unclear
access to KYCrypto is denied
End
customer pays the service fee to the bank
bank transmits wallet address and the corresponding bank ID to the oracle
oracle adds wallet address and the corresponding bank ID to its whitelist
End

Note. Own illustration.

9.4 Technical access to the smart contract

Customers who have successfully completed the onboarding process can now access the functions of our smart contract. From a technical point of view, there could be a front end for this, which has a connection to a wallet provider (e.g., MetaMask extension). Through this front end, the customer calls the functions (purchase, transfer, etc.) of our smart contract.

9.5 Purchase of KYCrypto

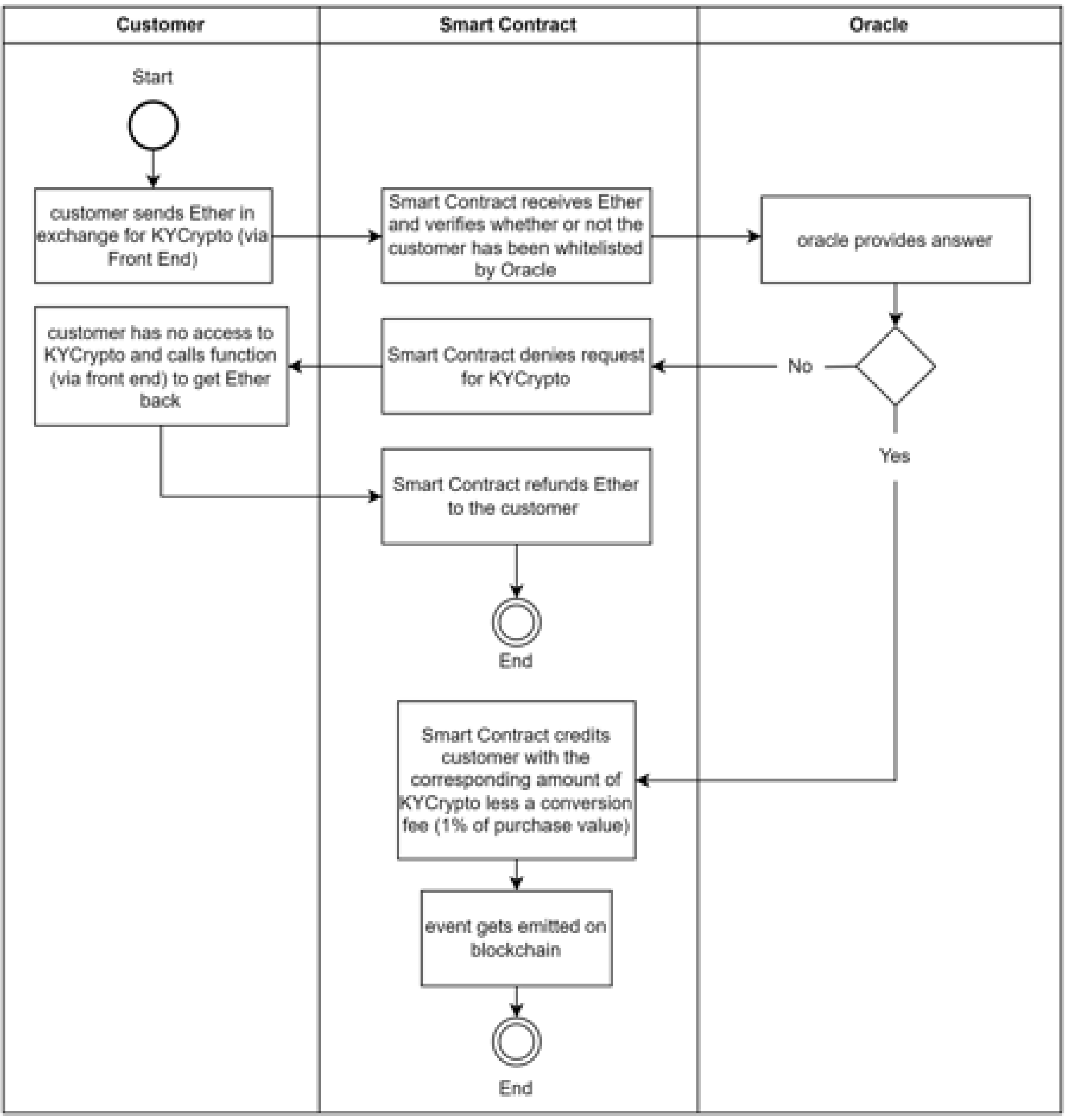
A customer who has already been onboarded by one of the participating banks and therefore has been whitelisted as stated above, meets the requirements of purchasing KYCrypto. Purchasing KYCrypto is only possible in exchange for ether. We deliberately chose ether as the exchange currency because our self-developed smart contract runs on the ethereum network. For this reason, the exchange currency must run on this network as well. Since ether is the main currency on the ethereum network, we have set ether as the exchange currency. However, the same principles could be applied in another blockchain network by adapting the smart contract to the corresponding coding environment. This allows us to set exchange currencies other than ether.

Subsequently, we describe step by step how a purchase of KYCrypto is carried out. In addition to that, we added below a figure which illustrates the detailed process. To purchase KYCrypto, the customer sends a request through the front end. This request goes to our smart contract. The smart contract automatically checks whether the purchase request was made from a whitelisted wallet address. For this purpose, the smart contract makes a request to the oracle to find out whether the corresponding wallet address has already been added to the whitelist.

In case the wallet address is not whitelisted, the smart contract sends the ether back to the customer. In contrast, if the wallet address has been whitelisted, then the customer receives KYCrypto against the sale of ether. The exchange of KYCrypto against ether takes place at a fixed exchange rate on a parity basis. As already mentioned in 9.2, a conversion fee of 1% is charged for each purchase and/or sale of KYCrypto. Most of the conversion fees are passed on to the banks, while a small portion of it is retained for the maintenance and operation of the IT infrastructure related to the smart contract, oracle, front end, and back end.

Figure 7

First purchase of KYCrypto


Customer
Smart Contract
Oracle
Start
customer sends Ether in exchange for KYCrypto (via Front End)
Smart Contract receives Ether and verifies whether or not the customer has been whitelisted by Oracle
oracle provides answer
customer has no access to KYCrypto and calls function (via front end) to get Ether back
Smart Contract denies request for KYCrypto
No
Yes
Smart Contract refunds Ether to the customer
End
Smart Contract credits customer with the corresponding amount of KYCrypto less a conversion fee (1% of purchase value)
event gets emitted on blockchain
End

Note. Own illustration.

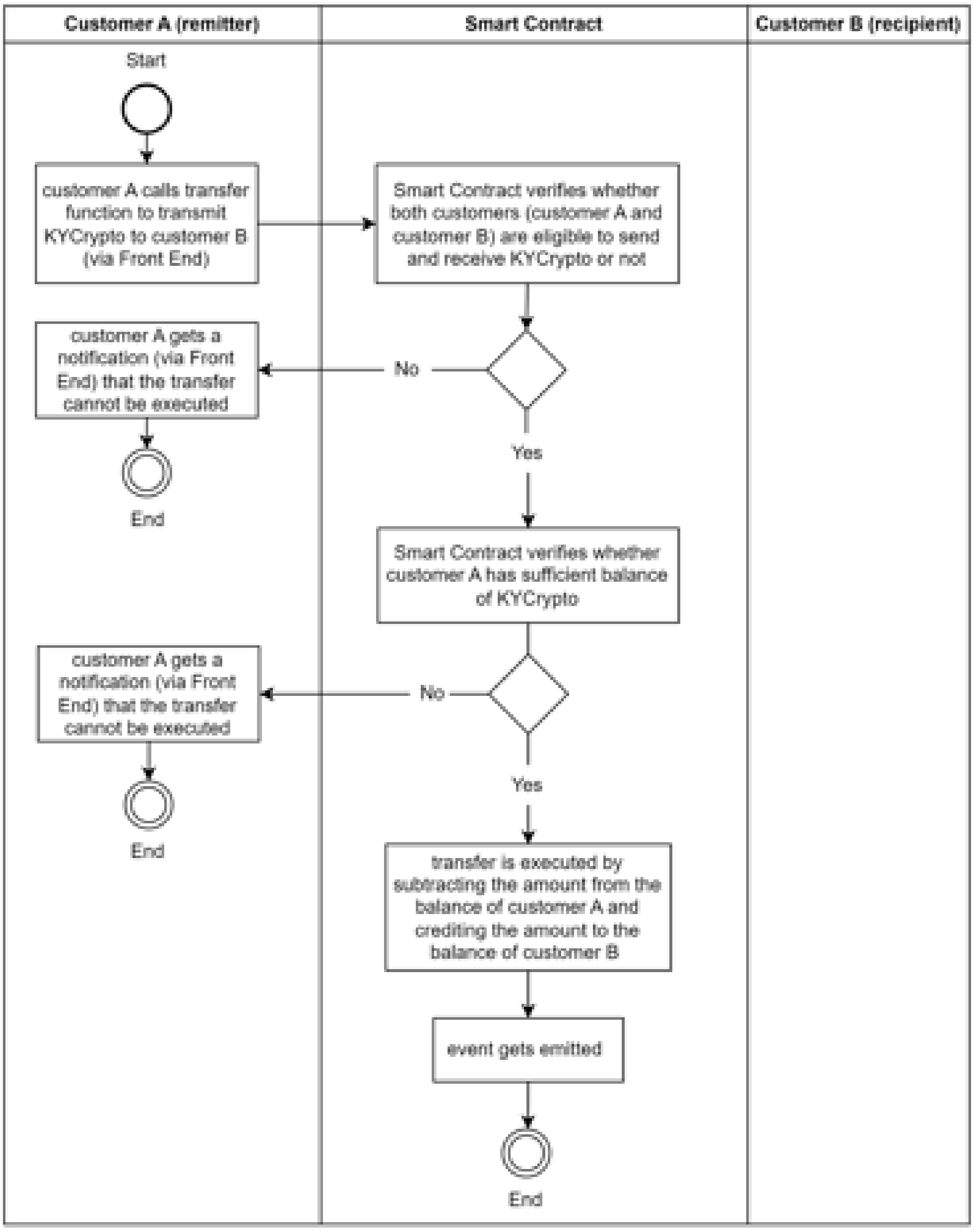
9.6 Transfer

As illustrated in the figure 8 below, the transfer of KYCrypto between two whitelisted customers is processed by the smart contract. Firstly, the remitter submits the transfer request to the smart contract via front end. Following that, the smart contract checks whether both customers, remitter and recipient, are eligible to send KYCrypto and receive KYCrypto. To be eligible, the customer must be whitelisted, and the corresponding wallet must not be blocked. Additionally, the smart contract checks whether the remitter has sufficient balance of KYCrypto to transfer the amount. If these requirements are fulfilled cumulatively, then the smart contract executes the requested transfer. Otherwise, the requested transfer cannot be executed.

Importantly, we made sure that the pseudo-anonymity is given when transferring ether (see also section 6.1) is also maintained in our concept. As already shown above, while onboarding the customers, the bank transmits not only the wallet address but also the corresponding bank ID to the oracle. Thus, the bank which carries out the onboarding process knows the identity of the newly admitted customer. However, the other banks do not have this information, as they do not know who is behind a third-party bank ID. Consequently, customers cannot identify each other in our concept either, which is why pseudonymity persists.

Figure 8

Transfer of KYCrypto


Customer A (remitter)
Smart Contract
Customer B (recipient)
Start
customer A calls transfer function to transmit KYCrypto to customer B (via Front End)
Smart Contract verifies whether both customers (customer A and customer B) are eligible to send and receive KYCrypto or not
No
customer A gets a notification (via Front End) that the transfer cannot be executed
End
Yes
Smart Contract verifies whether customer A has sufficient balance of KYCrypto
No
customer A gets a notification (via Front End) that the transfer cannot be executed
End
Yes
transfer is executed by subtracting the amount from the balance of customer A and crediting the amount to the balance of customer B
event gets emitted
End

Note. Own illustration.

9.7 AML and KYC

In this section, we analyze how AML and KYC-related processes are integrated to our KYCrypto concept. By doing so, we start with an analysis of our CIP and CDD during onboarding. Following that, we look into the way we conduct monitoring and then we explain how we deal with suspicious cases. Finally, we present our approach regarding KYC storage.

9.7.1 CIP and CDD during onboarding

For customers who do not yet have a bank account at the time of onboarding, the bank proceeds in the same way as when opening a conventional bank account from a regulatory perspective (see explanations in sections 7.3.1 and 7.3.3). This means that the bank carries out a CIP which is regarded as part of CDD. The customer therefore provides the bank with personal information (e.g., ID, domicile, job, etc.) and information about the financial situation (income, wealth, etc.).

As shown in Figure 6, for customers who already have a bank account at the time of onboarding, the bank also performs a CDD. However, this CDD is, as a matter of principle, less comprehensive than for the new customers since personal information and the financial situation of the customer is already available. The bank therefore checks whether this information is still up to date.

As shown in Figure 5 and Figure 6, opening a crypto wallet or providing an existing crypto wallet address belongs to the onboarding process. To complete the CDD, the bank verifies if the onboarding customer corresponds to the ultimate beneficial owner of the crypto wallet. In case wallet addresses have been used before, the bank additionally checks whether the transactions sent and received by this wallet are suspicious of money laundering.

If the bank notices anything suspicious when carrying out CDD, it reports this to the reporting office (see also our comments in section 7.4) according to art. 9 of AMLA (1997).

9.7.2 Monitoring

As shown in section 7.3.5, banks need to constantly make sure that they comply with the AML and KYC regulatory policies. In our solution, banks carry on with regularly monitoring of the underlying bank accounts as they do in the conventional system, as shown in section 7.3.5. Additionally, in our solution, the banks monitor the transactions of whitelisted wallets. From a technical view, banks analyze these transactions as they are processed on a public blockchain. Thus, details such as transfer amount, transfer date, etc., are publicly available on different block explorers such as Etherscan. Banks integrate this additionally collected transaction data into their existing monitoring software and analytical tools to monitor their customers on an ongoing basis.

As mentioned in section 7.3.3, there are four triggering events for performing a CDD according to the FATF (2023) recommendations. Besides establishing a business relationship as the first trigger event, the ongoing monitoring consists of three more triggering events that are significant at this stage. Those are as follows: occasional transactions exceeding threshold of 15'000 USD, suspicion of money laundering or terrorist financing, and outdated or incorrect information about customer identification.

As stated above, banks need to monitor the transactions constantly to detect these triggering events. This is why banks need to have organizational measures implemented to perform this monitoring systematically, as stated in section 7.3.2.

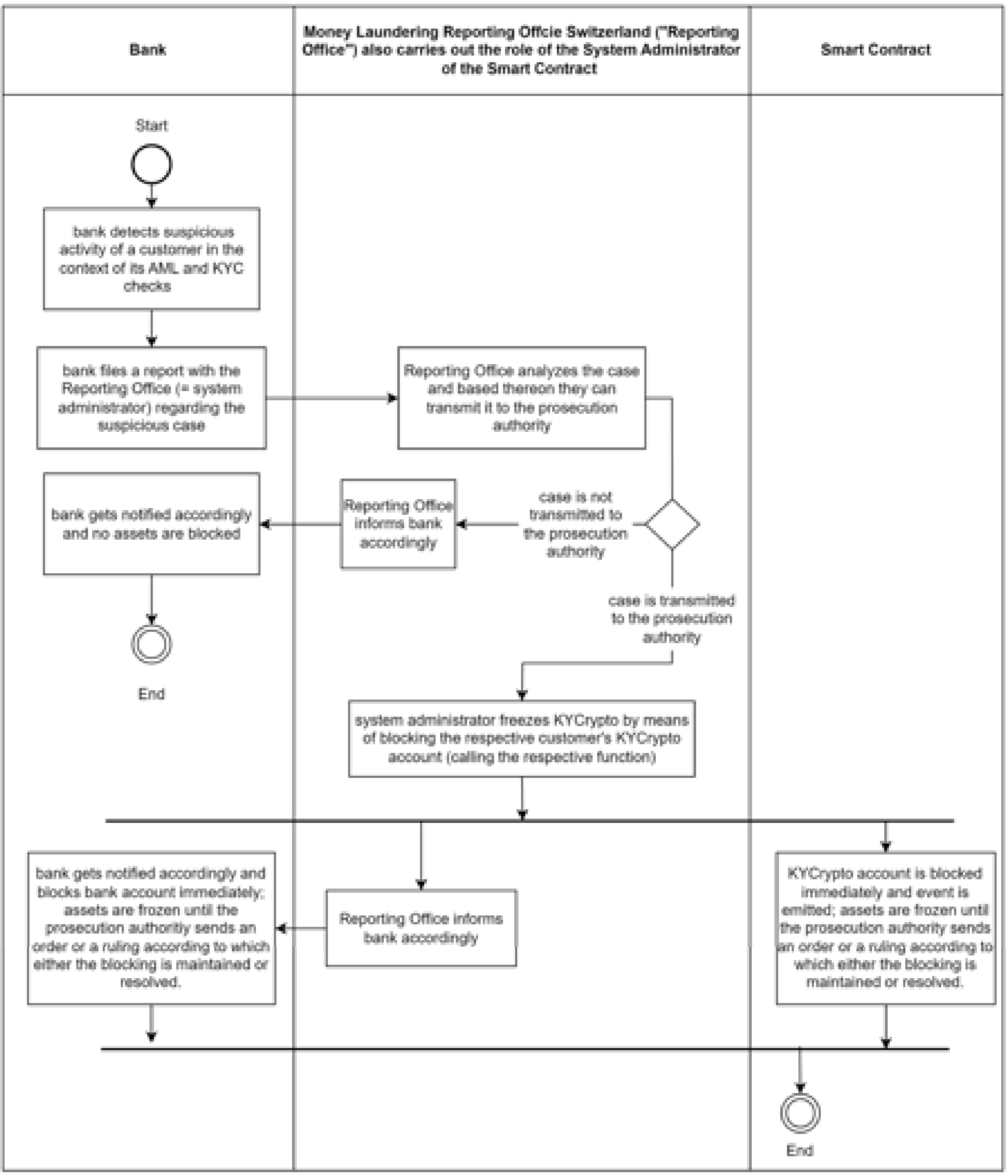
9.7.3 Handling in the event of a suspicious case

As analyzed in section 7.4, if a bank has reasonable grounds to suspect that a customer uses its bank account and/or its crypto wallet to launder money or to fund criminal activities, the bank needs to file a report with the reporting office (AMLA, 1997). Following that, the reporting office analyzes the case at hand. If there are sufficient and reasonable grounds that indicate suspicious activities, the reporting office transmits the case to the prosecution authority, whereby the bank is obligated to immediately freeze the assets on the suspicious bank account (AMLA, 1997).

In our solution, as already extensively discussed, there are not only conventional bank accounts involved but also KYCrypto accounts. This is the reason why we introduce a distinct process to block these KYCrypto efficiently. As shown in Figure 9 below, in our solution, the reporting office is concurrently the system administrator of our smart contract. If the reporting office forwards a case to the prosecution authorities, the reporting office can block the KYCrypto account itself by calling the corresponding function in the smart contract. This functionality of our self-developed smart contract ensures with immediate effect that the respective customer cannot withdraw KYCrypto tokens or Ether.

Figure 9

Blocking account


Bank
Money Laundering Reporting Offcie Switzerland ("Reporting Office") also carries out the role of the System Administrator of the Smart Contract
Smart Contract
Start
bank detects suspicious activity of a customer in the context of its AML and KYC checks
bank files a report with the Reporting Office (= system administrator) regarding the suspicious case
Reporting Office analyzes the case and based thereon they can transmit it to the prosecution authority
case is not transmitted to the prosecution authority
Reporting Office informs bank accordingly
bank gets notified accordingly and no assets are blocked
End
case is transmitted to the prosecution authority
system administrator freezes KYCrypto by means of blocking the respective customer's KYCrypto account (calling the respective function)
bank gets notified accordingly and blocks bank account immediately; assets are frozen until the prosecution authoritiy sends an order or a ruling according to which either the blocking is maintained or resolved.
Reporting Office informs bank accordingly
KYCrypto account is blocked immediately and event is emitted; assets are frozen until the prosecution authority sends an order or a ruling according to which either the blocking is maintained or resolved.
End

Note. Own illustration.

9.7.4 KYC storage

As described in section 7.3, according to the study by Pratz et al. (2021), compliance costs related to AML and KYC have increased significantly in recent years. According to Pratz et al. (2021), one of the cost drivers is the international lack of harmonized regulations, which makes it difficult for banks to simplify and optimize their KYC processes.

To achieve the goals of reducing compliance costs and enhancing the customer's experience, it is evident that scalable approaches such as utility services or data-sharing networks should be applied. However, according to Pratz et al. (2021) analysis, these have not yet become established. In section 8.1, we describe different ways to implement a blockchain-based data-sharing network. However, as already discussed in section 8.1, these blockchain-based solutions come with severe privacy issues. Therefore, we decided not to integrate a blockchain-based storage solution into our concept. In our concept, the KYC data is stored off-chain, either locally by banks or externally with a cloud service provider. As already stated above, we regard it as advantageous to have a utility service approach or a data-sharing network to reduce the costs incurred by the time-consuming collection of KYC data and thereby enhance the customer's experience. Furthermore, our opinion is that the regulators need to harmonize the AML and KYC policies internationally, which would facilitate such collaborative approaches.

To conclude, in our solution, we deliberately avoid limiting ourselves to only one of the above-stated approaches for storing and exchanging KYC data and instead allow the banks to choose the approach that suits them best.

9.8 Advantages of our concept

Our solution offers several advantages compared to the conventional system:

- With our concept and implementation solution, we achieve the goal of creating a self-developed token, KYCrypto, that is comparable to regulated as fiat money managed by a bank. In other words, with our implementation, we ensure that the existing AML and KYC-related regulatory processes are fully integrable. Our concept is a trust-building solution for institutional investors, as there is no other system in practice that could meet the same high regulatory requirements.

- Our solution includes a functional and tested smart contract that provides our KYCrypto token. The token is based on the ERC-20 token standard and thus allows the transfer of tokens and approval, as well as having control over the total supply of tokens available. Additionally, functions were added to provide additional options such as blocking accounts and withdrawing and depositing tokens.
- When a customer registers their wallet for the first time, the wallet address is automatically checked against the whitelist through the oracle, thus, creating an efficient process.
- The whitelist only includes wallet addresses, and the data of the customers is stored in individual banks. This provides anonymity from the other users while still providing the ability to enforce the AML and KYC-regulation. Additionally, because the data is stored in different banks, there is no centralized point of failure and the risk of every customer's data being breached is lower.
- Furthermore, we introduce a self-created business model that generates new revenue streams for banks in connection with their regulatory obligations. In addition, our blockchain-based smart contract helps to simplify transaction monitoring since the transactions in KYCrypto are publicly visible and, therefore, each bank can trace its customer behavior efficiently. This reduces compliance costs incurred by monitoring.
- Another advantage is that in the event of a suspicious case, not only can the wallet in question be blocked by the reporting office, but also related wallets that received or made these suspicious transactions in the KYCrypto environment. Until now, in the conventional system, the reporting office has to contact each bank separately to have the accounts blocked. This procedure can take a lot of time since counterparties can reside abroad, and the corresponding bank might be domiciled abroad as well. Consequently, our solution offers not only a strong simplification but also a significant acceleration in this regard.

To conclude, our focus is to develop a concept and implementation solution that builds on the current system. This allows banks to build on well-established regulatory processes. Consequently, no comprehensive reorganization on the part of the banks is necessary.

10. KYCrypto smart contract

To implement the concept described in the previous chapter, we created a smart contract. For its creation, our starting point was the code from former Blockchain Presence AG members Oliver Vertesi and Roman Willi who implemented certain features we make use of (Vertesi, 2022; Willi, 2021). Based thereon, we created a code that implements the ERC-20 token standard and additional features to ensure compliance with the regulatory frameworks for AML and KYC.

This chapter begins with an explanation of the ERC-20 token standard and how its functions are implemented in the smart contract, then the additional functions of the smart contract are outlined. Lastly, the chapter provides a brief overview of the testing procedures of the smart contract.

10.1 ERC-20 functions

The KYCrypto smart contract follows the ERC-20 standard, and in this paragraph, we briefly outline what ERC-20 is and what it signifies for our code. Subsequently, the different functions are introduced and the modifications that have been made to these functions are outlined.

ERC-20 is the ethereum standard for fungible tokens, introduced by Fabian Vogelsteller in 2015 (Ethereum, 2023). The standard can be adapted to fit specialized use, which makes it suitable for our implementation. Within ERC-20, there are different functions, some of which are optional while others are mandatory, and it also includes two events (Vogelsteller & Buterin, 2015). This provides the base functionality for our KYCrypto contract token. These base functions have been adapted so they fit with the current AML and KYC regulations. The ERC-20 token standard encompasses nine functions, out of which three are optional and six are mandatory functions. Of these nine functions, the six mandatory functions have been included in our code, which are as follows:

- *totalSupply*
- *balanceOf*
- *transfer*
- *transferFrom*
- *approve*
- *allowance*

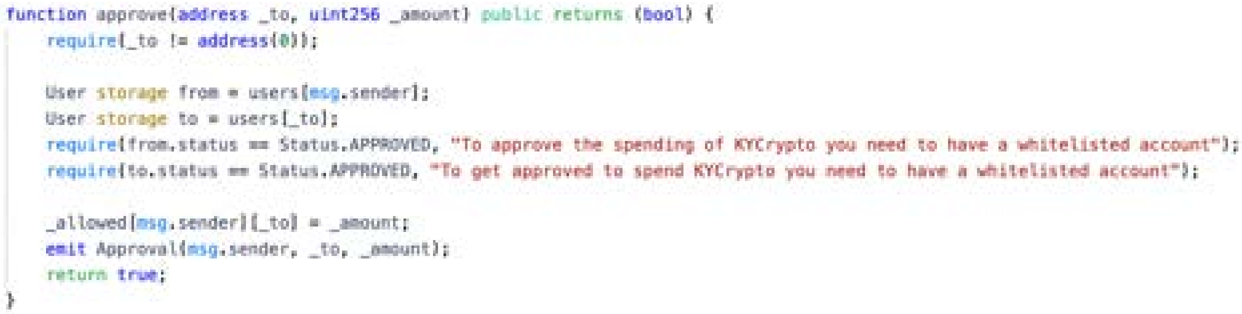
Additionally, we included functions that are not officially part of the ERC-20 standard but are closely connected to it. These functions are *Token_mint* and *Token_burn*, which allow modification of the *totalSupply* of tokens. Furthermore, *increaseApproval* and *decreaseApproval* enable adjustment of the amount that can be spent by the *approve* function.

Regarding the modifications of the functions, the different view functions like *totalSupply* and *allowance* have not been changed. Since there is no movement of funds or other state changes, there is no need to change the functions for compliance with the regulatory framework. The same applies to the functions responsible for minting and burning tokens: as the only way to get tokens in our system is to purchase them, there are no consequences for changing the total supply of tokens.

Changes were mainly made to the functions that handle transactions and approvals. The *approve* function allows another person to spend funds from another account (Vogelsteller & Buterin, 2015). Since these functions are involved in the transfer of money, certain precautions need to be taken. This primarily involves checking the status of the customers involved. The status of a customer is recorded in the *user struct* that differentiates between approved parties and not approved or not yet-approved parties. During a transaction or an approval process, all parties involved must be approved and listed on the whitelist of individuals screened by a bank. Checking the status of the customers is accomplished through simple require statements.

Figure 10

Approve function



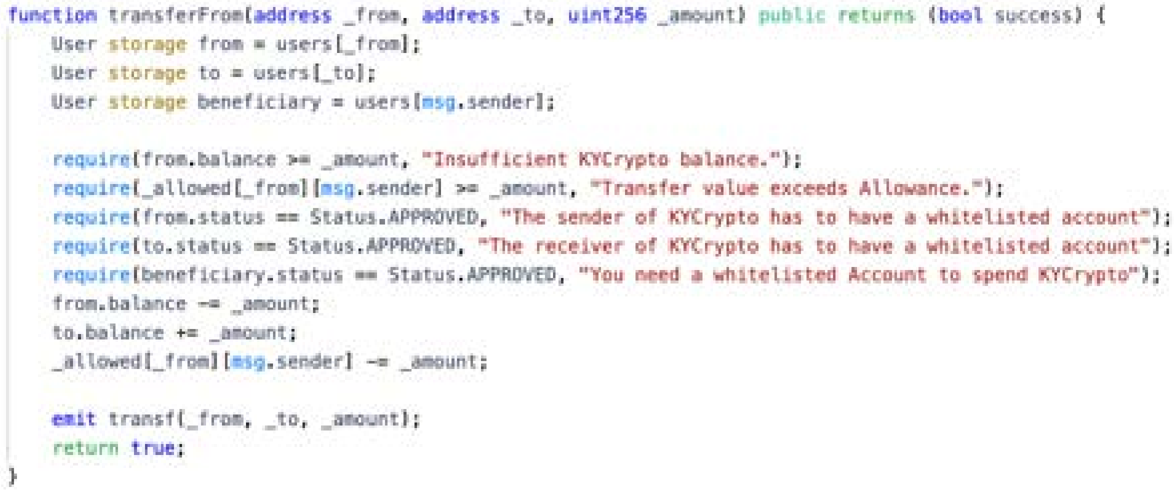
```
function approve(address _to, uint256 _amount) public returns (bool) {
    require(_to != address(0));

    User storage from = users[msg.sender];
    User storage to = users[_to];
    require(from.status == Status.APPROVED, "To approve the spending of KYCrypto you need to have a whitelisted account");
    require(to.status == Status.APPROVED, "To get approved to spend KYCrypto you need to have a whitelisted account");

    _allowed[msg.sender][_to] = _amount;
    emit Approval(msg.sender, _to, _amount);
    return true;
}
```

Note. Own illustration.

Before executing a transaction, the balance of the spender is checked to ensure they have enough tokens to complete the transfer. Additionally, when spending approved tokens, the number of spendable tokens is also verified. In the end, the actual transaction is a simple addition for one party and a subtraction for the other party.

Figure 11

Transfer function



```
function transferFrom(address _from, address _to, uint256 _amount) public returns (bool success) {
    User storage from = users[_from];
    User storage to = users[_to];
    User storage beneficiary = users[msg.sender];

    require(from.balance >= _amount, "Insufficient KYCrypto balance.");
    require(_allowed[_from][msg.sender] >= _amount, "Transfer value exceeds Allowance.");
    require(from.status == Status.APPROVED, "The sender of KYCrypto has to have a whitelisted account");
    require(to.status == Status.APPROVED, "The receiver of KYCrypto has to have a whitelisted account");
    require(beneficiary.status == Status.APPROVED, "You need a whitelisted Account to spend KYCrypto");
    from.balance -= _amount;
    to.balance += _amount;
    _allowed[_from][msg.sender] -= _amount;

    emit transf(_from, _to, _amount);
    return true;
}
```

Note. Own illustration.

10.2 Additional functions

With the implementation of ERC-20 functions, the base functionality of our token is set. Nevertheless, to attain full compliance with the regulatory guidelines, further features need to be integrated. These features include registration, conversion of funds, blocking and unblocking accounts. To accomplish this, the following functions are incorporated additionally:

- *_Account_Registration*
- *_depositKYCrypto*
- *Mailbox*
- *_withdrawthedeclinedCrypto*
- *_depositKYCrypto*

- *_withdrawKYCrypto*
- *Account_Blocking*
- *Account_Unblocking*

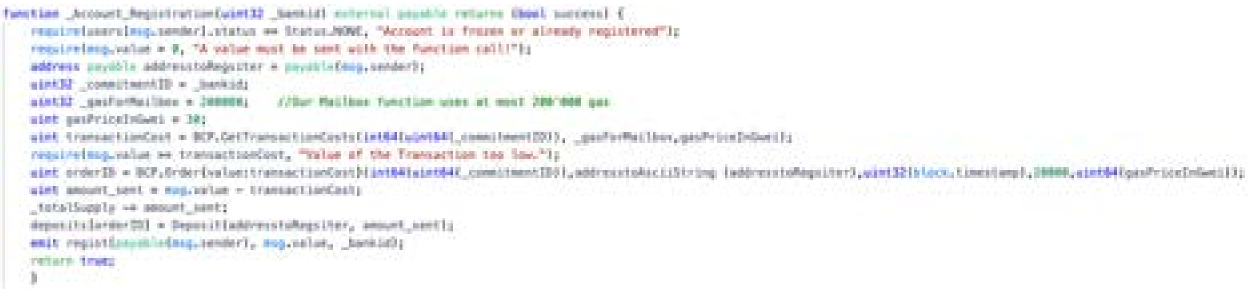
In the following subchapters, we elaborate on different functions ad in detail, starting with account registration and the functions needed. Subsequently, we explain the functions for depositing and withdrawing funds, and in the end, we examine of the functions for blocking and unblocking accounts.

10.2.1 Account registration

The *_Account_Registration* function is one of the most important functions in the smart contract as it sends the request to the oracle. Initially, the function checks that the wallet address of the user is not yet registered. It is important that an existing account does not call this function because that would result in the transaction costs being paid multiple times to the oracle (to the financial detriment of the customer). Even worse, the status of the account may be overwritten (a significant problem if it happens to a blocked account), which we will discuss later. In a next step, the transaction costs for sending a request to the oracle are determined, and it is checked that the value sent with the function is higher than that. The rest of the money that is sent is stored in the *Deposit struct* for use when the oracle request is returned.

Figure 12

Account registration function



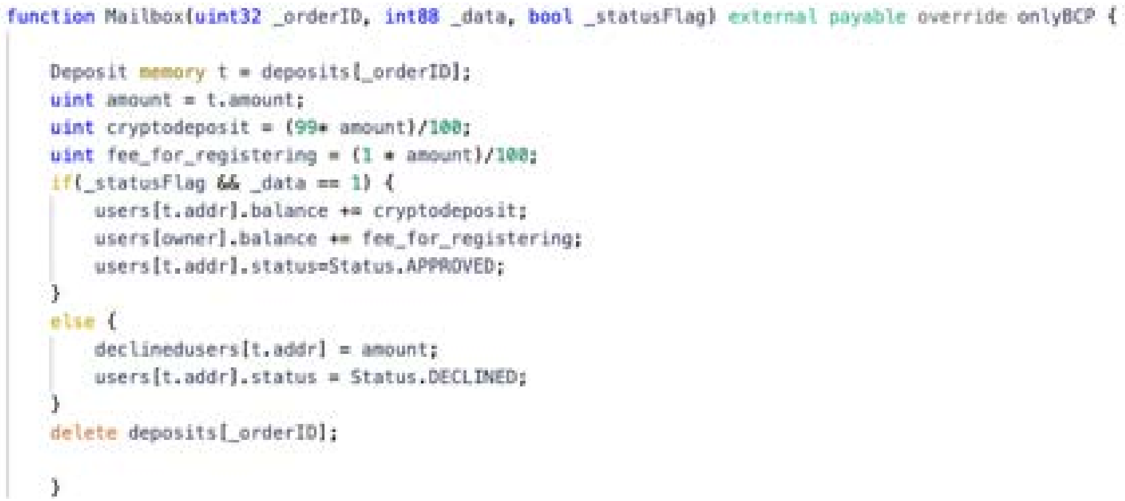
```
function _Account_Registration(uint32 _bankid) external payable returns (bool success) {
    require(users[msg.sender].status == Status.NONE, "Account is frozen or already registered");
    require(msg.value > 0, "A value must be sent with the function call!");
    address payable addresstoRegister = payable(msg.sender);
    uint32 _commitmentID = _bankid;
    uint32 _gasForMailbox = 200000;    //Our Mailbox function uses at most 200'000 gas
    uint gasPriceInGwei = 30;
    uint transactionCost = BCP.GetTransactionCosts(int64(uint64(_commitmentID)), _gasForMailbox,gasPriceInGwei);
    require(msg.value >= transactionCost, "Value of the Transaction too low.");
    uint orderID = BCP.Order{value:transactionCost}(int64(uint64(_commitmentID)),addresstoAsciiString (addresstoRegister),uint32(block.timestamp),20000,uint64(gasPriceInGwei));
    uint amount_sent = msg.value - transactionCost;
    _totalSupply += amount_sent;
    deposits[orderID] = Deposit(addresstoRegsiter, amount_sent);
    emit regist(payable(msg.sender), msg.value, _bankid);
    return true;
    }
```

Note. Own illustration.

After the oracle has checked the account address against the whitelist, it sends the information back to the smart contract. The smart contract receives this information in the mailbox function. In the current version of the smart contract, the information sent only differentiates between being on the whitelist and not being on the whitelist. The information is sent as an int88, which means that a positive or negative number of up to 88 bits can be sent (Amanwachi, 2022). Possibly, additional information like risk category or information about the customer could be sent as well. However, we do not make use of this as the transaction monitoring takes place off-chain and the information might interfere with the semi-anonymity attribute. Referring back to the smart contract, if the person is on the whitelist, the conversion fee is subtracted and the money that is stored in the deposit struct is transferred to the customer's balance. Additionally, the status is set to *approved,* meaning the customer is now freely able to use their tokens. On the other hand, if someone is not found on the whitelist, their status is set to *declined*. The amount of money that they paid is transferred into the *declinedusers struct* and can then be claimed through the *_withdrawthedeclinedCrypto* function. This function also resets the person's status.

Figure 13

Mailbox function

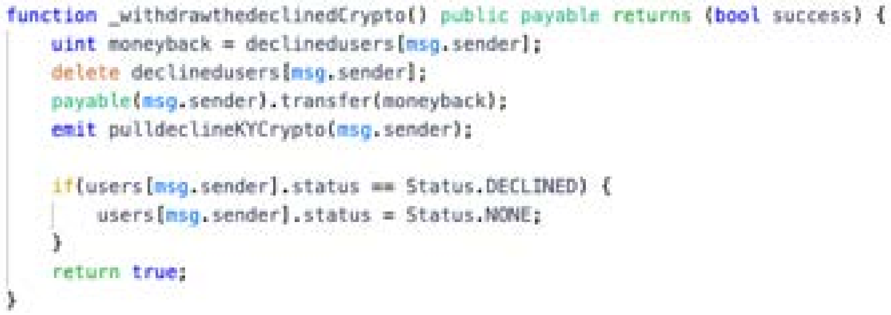


```
function Mailbox(uint32 _orderID, int88 _data, bool _statusFlag) external payable override onlyBCP {

    Deposit memory t = deposits[_orderID];
    uint amount = t.amount;
    uint cryptodeposit = (99* amount)/100;
    uint fee_for_registering = (1 * amount)/100;
    if(_statusFlag && _data == 1) {
        users[t.addr].balance += cryptodeposit;
        users[owner].balance += fee_for_registering;
        users[t.addr].status=Status.APPROVED;
    }
    else {
        declinedusers[t.addr] = amount;
        users[t.addr].status = Status.DECLINED;
    }
    delete deposits[_orderID];

    }
```

Note. Own illustration.

Figure 14

Function for withdrawing declined KYCrypto



```
function _withdrawthedeclinedCrypto() public payable returns (bool success) {
    uint moneyback = declinedusers[msg.sender];
    delete declinedusers[msg.sender];
    payable(msg.sender).transfer(moneyback);
    emit pulldeclineKYCrypto(msg.sender);

    if(users[msg.sender].status == Status.DECLINED) {
        users[msg.sender].status = Status.NONE;
    }
    return true;
}
```

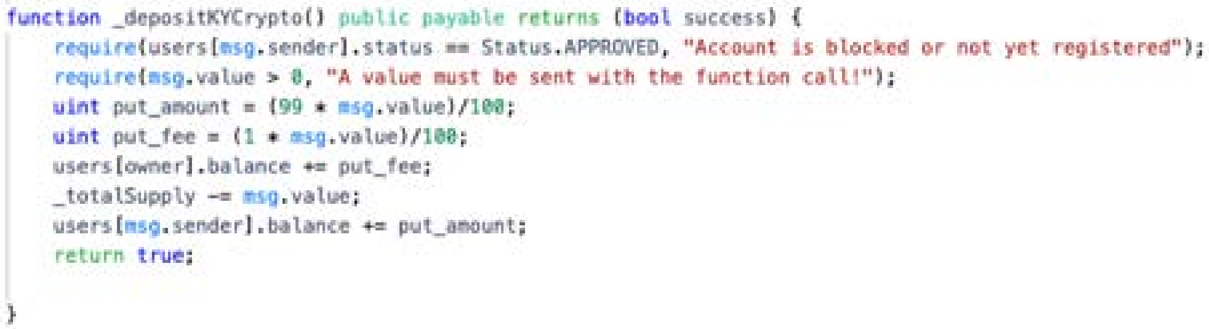
Note. Own illustration.

10.2.2 Depositing and withdrawing funds

For users that already possess an account and want to deposit more tokens into their account, there is the *_depositKYCrypto* function. This function first checks if the customer is already approved, so this function cannot be used to circumvent the *_Account_Registration* function. Furthermore, the conversion fee is once again subtracted from the value, and the rest is added to the balance of the customer.

Figure 15

Deposit function



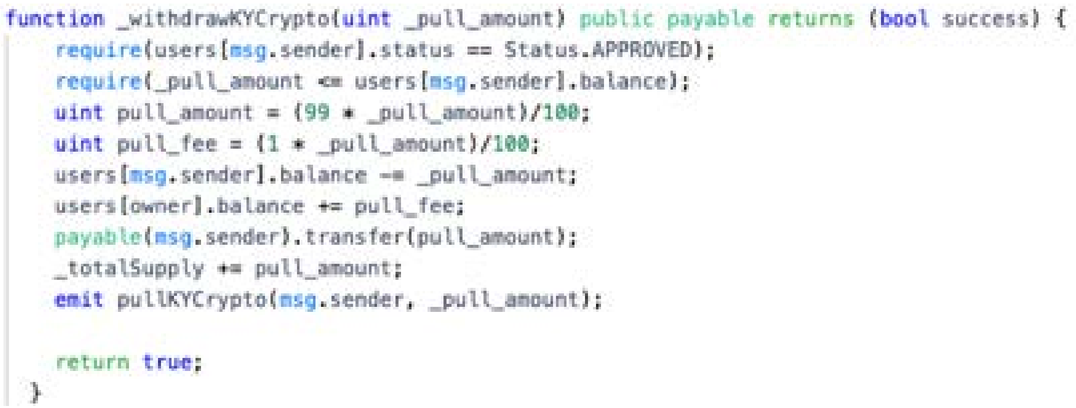
```
function _depositKYCrypto() public payable returns (bool success) {
    require(users[msg.sender].status == Status.APPROVED, "Account is blocked or not yet registered");
    require(msg.value > 0, "A value must be sent with the function call!");
    uint put_amount = (99 * msg.value)/100;
    uint put_fee = (1 * msg.value)/100;
    users[owner].balance += put_fee;
    _totalSupply -= msg.value;
    users[msg.sender].balance += put_amount;
    return true;

}
```

Note. Own illustration.

If a customer wants to withdraw his tokens and turn them back into ether, the *_withdrawKYCrypto* function comes into play. The first step of this function is to check the status of the customer, making sure that the account is approved. The second step is to check the requested withdrawal amount of the token to ensure it does not exceed the customer's token balance. If the two conditions are met, the amount that the customer wants to withdraw is subtracted from his KYCrypto account, and the corresponding amount of ether (apart from the 1% conversion fee) is transferred to the customers' wallet address.

Figure 16

Withdraw function



```
function _withdrawKYCrypto(uint _pull_amount) public payable returns (bool success) {
    require(users[msg.sender].status == Status.APPROVED);
    require(_pull_amount <= users[msg.sender].balance);
    uint pull_amount = (99 * _pull_amount)/100;
    uint pull_fee = (1 * _pull_amount)/100;
    users[msg.sender].balance -= _pull_amount;
    users[owner].balance += pull_fee;
    payable(msg.sender).transfer(pull_amount);
    _totalSupply += pull_amount;
    emit pullKYCrypto(msg.sender, _pull_amount);

    return true;
 }
```

Note. Own illustration.

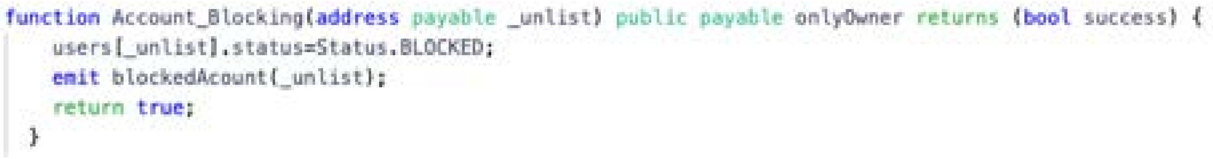
10.2.3 Blocking and unblocking of accounts

As was previously stated, our token contract can block accounts with the *Account_Blocking* function. This function can only be used by the owner of the contract (in our case, a System Administrator). The status of the customer is designated as blocked, and this means that the account is not able to use most of the functions in our contract. This includes transfers, approvals, transferring tokens from another account or having money transferred from the blocked account through another person, as well as the ability to deposit and withdraw money. This is done to freeze an account in compliance with the AML regulation, which requires an account to be frozen in certain situations, namely if the bank has filed a report with the money laundering reporting office

Switzerland after art. 9 of the AMLA and has been informed that the case has been given to a prosecuting authority (AMLA, 1997).

Figure 17

Account blocking function



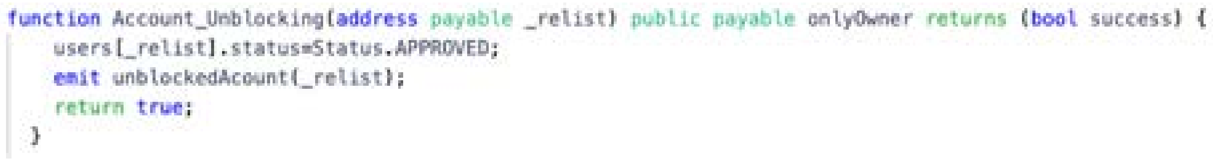
```
function Account_Blocking(address payable _unlist) public payable onlyOwner returns (bool success) {
    users[_unlist].status=Status.BLOCKED;
    emit blockedAcount(_unlist);
    return true;
 }
```

Note. Own illustration.

The ability to unfreeze or unblock the account is provided by the *Account_Unblocking* function. This function sets the status of the customer back to approved and, therefore, allows the utilization of all the functions in the contract. This function is necessary in the case that the customer is ruled innocent by the prosecuting authority or if the maximum time for which an account can be frozen has expired. In art. 10 of the AMLA, it is defined that the maximum amount of time that an account can be frozen is five working days after the transmission of the reporting office (AMLA, 1997).

Figure 18

Account unblocking function



```
function Account_Unblocking(address payable _relist) public payable onlyOwner returns (bool success) {
    users[_relist].status=Status.APPROVED;
    emit unblockedAcount(_relist);
    return true;
 }
```

Note. Own illustration.

10.3 Testing

While building the smart contract, it clarified that the functionality of the smart contract would need to be tested. Due to the fact that this paper does not aim to create an oracle, we turned to Florian Rüegsegger. He created a fake oracle for us to test the smart contract.

The fake oracle is a smart contract that enables us to mimic the correspondence with the oracle by allowing us to call a function that sends information to the smart contract in the same form that the oracle would. With the fake oracle the different functions and eventualities could be tested out, helping to identify and rectify any bugs or errors.

The fake oracle, as well as a guide for testing the smart contract using the fake oracle, are included in the appendix. Additionally, the bcp_informed smart contract is also listed in the appendix, so that all the parts of code used are available for recreation purposes.

11. Results and key findings

KYCrypto, as the result of the described approach in the method section, established a solid foundation for SCL students and researched toward the intended goals. To elaborate on our results and findings, this chapter presents a comprehensive overview of the previous two chapters. Our primary objective in this chapter is to synthesize the information and outline the components that contributed to KYCrypto's regulated environment. After clarifying this, we highlight our key findings.

The KYCrypto concept and KYCrypto smart contract are mutually dependent and should therefore be understood as an overall result of this paper. The concept explained the processes related to AML and KYC along with our smart contract. Through activity diagrams, based on the unified modeling language, we showed in our concept how different on-chain and off-chain processes worked by creating different paths and endings. Thereby, our smart contract and the oracle interactions were involved in several steps. Further, we differentiated the different involved parties that interacted with each other. As a result, we illustrated responsibilities, as well as the dependencies between the parties, to gain a comprehensive understanding of the entire KYCrypto process. In addition, our smart contract integrated on a theoretical basis the bank's proven AML and KYC-related regulatory processes. This implies that the smart contract-based cryptocurrency, KYCrypto, could be as strictly regulated as conventional bank-managed money.

KYCrypto has four crucial parts to create an efficient environment for regulation. The first part is to prevent fraudulent customers from entering the regulated environment by conducting CIP and CDD, as explained in the KYCrypto processes. Consequently, we developed the KYCrypto smart contract in such a way that it sends a request to the oracle for each new customer to confirm whether they have been whitelisted or not. The second part enables customers to do transactions through the smart contract. Therefore, we added usual payment processing tasks into the smart contract. By adding these tasks, a customer can exchange ether for KYCrypto tokens. These tokens can be sent, received, and exchanged back into ether. The third part is to track the transactions. By tracking transactions, investigating entities can check if the customers act suspiciously after they entered the regulated environment. The fourth part is to prevent transactions from happening by blocking suspicious customers. As a result, the KYC elements CIP, CDD, and transaction monitoring are implemented in the KYCrypto concept and within the smart contract. Further, AML prosecution is also feasible, creating an approach to regulate the blockchain environment.

One key finding in the KYCrypto process is at the customer onboarding, where banks perform critical regulatory measures in CIP and CDD. Our onboarding process aims to whitelist only the customers who do not raise any suspicion, and therefore, we identified that different customers needed to be treated differently in our processes. For instance, if a customer does not have an existing business relationship or a crypto wallet, useful transaction history may be missing. Consequently, we adapted our concept to this.

Another key finding is related to the monitoring of KYCrypto transactions and identifying criminal customers. We argue that monitoring and identification could be done more sophisticatedly than monitoring conventional money because, with our smart contract, every entity could always be up to date about every transaction. Therefore, it is very transparent since all previous transactions can be traced back through different investigating entities by using etherscan. Furthermore, if a customer is identified as suspicious, it may be possible to trace back the previous transactions and identify more suspicious customers. This may lead to the ability to identify and freeze assets of entire criminal organizations. Furthermore, this implies that criminal actions off-chain may have an effect on-chain and vice versa. As a result, monitoring of KYCrypto transactions could lead to a paradigm shift for AML prosecution.

12. Discussion

In this chapter, we examine the connection between KYCrypto and our literature review chapter. We will critically discuss both the similarities and differences between KYCrypto and the current literature. Furthermore, we highlight the unique contributions and the added values that KYCrypto brings to the domain. Additionally, we critically discuss related topics to the literature and explore the limitations of KYCrypto. This chapter is divided into four sub-chapters: KYC data storage, transaction monitoring, data privacy, and the smart contract.

12.1 A critical analysis of KYC data storage

After analyzing the PRISMA literature review of Malhotra et al. (2022), we recognized the significance of blockchain storage solutions. Parra-Moyano and Ross (2017) discussed blockchain KYC processes and data storage, while Singhal et al. (2020) expanded on this topic by implementing IPFS. Whether centralized or decentralized, we think that both approaches are very interesting and could synergize with our concept. However, we consider implementing it only if the data privacy is aligned with the standards, the technology is secure, and the financial companies are not concerned about reverse engineering (Parra-Moyano & Ross, 2017). Due to these open concerns, KYCrypto does currently not involve storing sensitive customer data on the blockchain. Therefore, we do not store either a hash of KYC data on a distributed ledger or use IPFS for cryptographic linking to a file. To acknowledge the importance of data storage solutions, examined in the AML and KYC within a regulatory framework chapter, we researched existing KYC storage solutions of traditional approaches. According to Pratz et al. (2021), there are four relevant ones: the bank-internal optimization, managed service, utility service, and the data-sharing network approach. By using a more traditional approach to storing KYC data, we wanted to allow financial companies to utilize their existing storage solutions. This approach adds significant value to our paper as KYCrypto aims to be implemented through familiar infrastructure and processes, aligning with practices in financial institutions.

The inability to share sensitive customer data through the blockchain imposes a limitation on KYCrypto. Linking complete customer data packages to their wallet address would significantly improve data-gathering efficiency and speed, in our opinion. It would enhance the assessment of customer behavior in detecting suspicious activity and provide an enhanced understanding of initially unexplained transactions, enabling more accurate identification of peculiar transactions. Further, if a whitelisted KYCrypto customer submits a suspicious transaction and gets reported by the bank to the regulating authority, the regulators may have already access to AML and KYC-related data. In case it is a criminal transaction, we think that the customer could be blocked faster from the whitelist and immediately pursued by the law. However, to our current understanding, there is no well-established blockchain-based solution for KYC data storage. However, the traditional system has already established processes for the storage and exchange of KYC data, as described by Pratz et al. (2021). Therefore, KYCrypto is aimed to rather use these instead of blockchain-based solutions.

12.2 A critical analysis of transaction monitoring

The literature review provided valuable insights into existing research on blockchain technology and AML practices. Oad et al. (2021) proposed a blockchain-enabled transaction scanning method. The goal of this method is to identify anomalous actions in financial transactions. While their approach gives us interesting insights, we recognize the complexity of defining outliers since customers have diverse backgrounds. Further setting rules to identify deviations may change over time since customers earn changing amounts over a business relationship. As a result, the on-chain implementation seems complex. Similarly, Xu et al. (2021) presented a blockchain-based AML system for the CDD process, which focuses on the secure storage of customer data and smart contracts automation. Their real-time updates and smart contract automation align with our objectives. However, legal problems could arise when integrating standardized information among different financial institutions. The last approach to detect money laundering activities, that we presented in our literature review, was from Singh and Best (2019) who use visualization techniques. Their approach can identify money laundering within a bank, but it cannot trace money flows across multiple entities. However, nowadays, we must consider that customers typically have more than one account, and these are across various banks. After reviewing these papers, we argue that banks already have a quite advanced transaction monitoring systems, therefore we decided that it would be more effective to leverage existing infrastructure than developing a new monitoring system.

In our approach there are also limitations, one of them being the reliance on the bank's compliance work, as we assume that each bank will conduct KYC processes diligently. Secondly, we also recognize a limiting factor in the transaction analysis of crypto wallets during the onboarding process. This review can only be carried out to a limited extent because the counterparties, as a matter of principle, may not be identified easily. However, this is not a disadvantage compared to the conventional system because certain transactions, especially cash transactions, cannot be exactly traced either. Considering these limitations, it becomes evident that the detection of money laundering has its complexity and ongoing improvement is required.

12.3 A critical analysis of data privacy

Referring to the literature review, in our opinion KYCrypto is comparable with the privacy-preserving KYC exchange of Biryukov et al. (2018). Likewise, they explicitly used a whitelist and a KYC implementation in the blockchain environment. Furthermore, they created a use case with two basic figures that can be compared to our concept part and created a proof-of-concept implementation that can be compared to our smart contract. In contrast to our approach, they focused more on the privacy aspects, and we focused more on the KYC process. In addition, their approach is more about creating a crypto exchange, whereas our approach is about regulating cryptocurrencies. As a result, our contribution to the current literature is, on the one hand, an approach that is more focused on regulating cryptocurrencies and, on the other hand, creating a holistic concept that explains the steps in detail.

As we use in our concept a whitelist as well, the question came up whether it is also relevant for us to implement a cryptographic accumulator or not. In the literature review, we assumed that there is only one KYC-conducting company that controls the onboarding and the transactions of customers. But in our opinion, in a payment system, there should be more than one KYC-conducting company because scalability issues, governance issues, and country borders could be limiting factors. But if it is still possible to conduct the KYC process with more than one conducting company, the privacy feature based on the cryptographic accumulator of Camenisch et al. (2009) should be implemented.

After analyzing the similarities and distinctions of Biryukov et al. (2018), an important limiting factor to consider is the semi-anonymous approach of KYCrypto. While customers seeking more anonymity may prefer the privacy aspect of KYCE, it should be noted that complete anonymity hampers effective AML prosecution. KYC investigators desiring full transparency may also be dissatisfied due to the presence of semi-anonymity. Investigators can only see the bank ID and wallet address, with the account holder's name. This prevents in-house investigators from linking wallet addresses to customers of other banks easily. However, it is crucial to highlight that the own customer of a bank should be fully identifiable. Therefore, if each bank conducts proper investigations, the regulated environment can remain efficient. In case, there are suspicious transactions from a customer of another bank, investigators can still identify the bank ID linked to the wallet address. Consequently, they can reach out to the other bank for more information or report it directly to the AML reporting office. Hence, the investigator's ability to act could remain unrestricted. Although semi-anonymity limits the effectiveness of AML prosecution and reduces system transparency, it mitigates privacy concerns compared to a complete lack of anonymity. For this reason, our concept is a semi-anonymous approach, whereas we assume that the approach of Biryukov et al. (2018) is more anonymous.

12.4 A critical analysis of the smart contract

One of the added values of our smart contract compared to existing research approaches, such as Biryukov et al. (2018), is the focus on the procedural aspects of AML and KYC. Our smart contract also offers the possibility to verify and block an account. Furthermore, in contrast to many other existing research approaches, such as Parra-Moyano and Ross (2017), we developed the smart contract in such a way that it creates a request to the oracle for each new customer to check whether they are already whitelisted. Compared to existing research approaches, we considered how to generate new revenue sources for the banks. We achieved this by implementing conversion fees directly in the smart contract. Therefore, we make sure that banks are compensated for the compliance expenses of their AML and KYC-related IT infrastructure.

However, our smart contract also has some limitations. One of the concerns is a blockchain's inherent property of immutability. This means that once a smart contract is uploaded, as a matter of principle, it cannot be changed (Hewa et al., 2021). If there are vulnerabilities in the code, this could be exploited by hackers (Wang et al., 2018). Fixing this issue may require a lot of effort and lead to high financial costs (Hewa et al., 2021).

Another limitation is that we did not implement any blockchain-based transaction monitoring to detect any anomalies. Such blockchain-based transaction monitoring schemes have already been introduced by Oad et al. (2021), Xu et al. (2021), and Singh and Best (2019), as shown in our literature review. Another limiting aspect is that the data on the smart contract is not constantly updated by the oracle. Therefore, if an account needs to be blocked immediately, the reporting office must do it manually by calling the blocking function on the smart contract.

Another limitation of our solution is that we implemented a single exchange currency, namely ether. Thus, we linked the value of KYCrypto directly to ether. This naturally entails fluctuation risks, as ether is exposed to the market. However, by applying the mint and burn functions in our self-developed smart contract, we could mitigate this fluctuation risk. An alternative is the use of stablecoins instead of ether as the exchange currency.

13. Conclusion

In the following, we address our conclusions about the research questions presented at the beginning of this paper. We also show other research fields that are of interest for future work.

The first part was about analyzing the business processes of account opening and payment processing. We discovered that in the conventional system, opening accounts often still requires physical presence and incurs significant costs for banks. In contrast, the process of opening a crypto wallet is more efficient. However, it is important to note that crypto wallets raise serious security concerns. In terms of transaction processing, we identified three main differences between the conventional and the blockchain-based system. Firstly, in the conventional system, transactions are processed in a centralized manner, whereas in the blockchain-based system, they are processed on a decentralized basis. Furthermore, blockchain-based transactions are irreversible, while conventional transactions can be reversed as a matter of principle. Lastly, blockchain-based transactions demonstrate faster processing times compared to conventional transactions.

Regarding the analysis of AML and KYC regulations in the Swiss context, we figured out that these provisions are made up of an international and national framework of guidelines, recommendations and laws. Our paper shows how these different sources interact. Additionally, our paper shows how this comprehensive regulatory framework is implemented by the banks.

Our paper demonstrates the existence of a wide range of research approaches aimed at developing blockchain-based KYC storage and exchange solutions. Some approaches focus on transferring data by creating a hash that is stored either centrally or by using IPFS, others try to integrate a blockchain-based whitelist that is encrypted by a cryptographic accumulator. Many other approaches exist. However, it is worth noting that, overall, no single approach has yet established itself in practice.

To integrate the AML and KYC-related regulatory processes into a cryptocurrency, we created a whitelist connected to our blockchain-based smart contract. This approach helps mitigate money laundering on the blockchain by aiming to provide tokens exclusively to whitelisted customers. By leveraging the transparency of blockchain technology and combining it with AML and KYC processes, it enhances the trustworthiness and legitimacy of cryptocurrencies, laying a foundation for wider acceptance in the current financial landscape.

On top of that, we introduced conversion fees and service fees, which enable banks to generate revenues from its AML and KYC compliance services. We designed our smart contracts in a way that the central functions of our conceptual solutions (e.g., conversion fees, requests to the oracle, or blocking) can be executed directly on the smart contract. This automated forwarding of fees serves as an additional incentive for traditional financial institutions to engage in AML and KYC of digital assets.

Two primary research areas emerge as potential fields for further research. Firstly, the field of KYC data storage and exchange presents opportunities for investigation. In our conceptual solution, we did not implement KYC data storage and KYC data exchange on the blockchain, as we opted to remain off-chain to prioritize privacy and security in this regard. However, there are many research approaches that demonstrate various methods of securely storing KYC data on the blockchain and making it accessible. Consequently, there is potential to expand our concept in this regard. The second field of research that we recommend addressing in the future concerns the choice of exchange currency. In our concept, we directly linked our currency, KYCrypto, to the value of ether. Nonetheless, as mentioned in the discussion section, this poses a significant risk due to exchange rate fluctuations. Therefore, it would be beneficial to consider an alternative reference currency, such as stablecoins, which are linked to a conventional currency, in the future. Our smart contract could be modified accordingly.

Authors

Jasmin Reber

Publication

Valentin Leuthard

Operations

Oliver Erni

Management

Alex Weber

Operations

Ann-Sophie Pfammatter

Publication

References

Acharya, P. (2021, November 24). *Domestic Payment Schemes in the UK – Part 1.* Payments Domain. https://www.paymentsdomain.com/2021/11/24/domestic-payment-schemes-in-the-uk-part-1/

Agreement on the Swiss banks' code of conduct with regard to the exercise of due diligence, (CDB 20), (2020). https://www.swissbanking.ch/_Resources/Persistent/6/2/e/e/62eec3df0685e359c5a376dfca79dec8b908ea9c/SBA_Agreement_CDB_2020_EN.pdf

Akbar, N. A., Muneer, A., ElHakim, N., & Fati, S. M. (2021). Distributed Hybrid Double-Spending Attack Prevention Mechanism for Proof-of-Work and Proof-of-Stake Blockchain Consensuses. *Future Internet*, *13*(11), 285. https://doi.org/10.3390/fi13110285

Alkhalili, M., Qutqut, M. H., & Almasalha, F. (2021). Investigation of Applying Machine Learning for Watch-List Filtering in Anti-Money Laundering. *IEEE*, *9*, 18481–18496. https://doi.org/10.1109/ACCESS.2021.3052313

Amanwachi, T. (2022, April 29). *The ultimate guide to data types in Solidity.* LogRocket Blog. https://blog.logrocket.com/ultimate-guide-data-types-solidity/#solidity-value-types

Apathy, P., Iro, G., & Koziol, H. (1993). *Österreichisches Bankvertragsrecht [Austrian Banking Contract Law]: Band 2: Konto und Depot [Volume 2: Account and deposit]*. Springer-Verlag/Wien.

Arnold, M. (2018, June 6). Ripple and Swift slug it out over cross-border payments. *Financial Times*. https://www.ft.com/content/631af8cc-47cc-11e8-8c77-ff51caedcde6

Azouvi, S., Goren, G., Heimbach, L., & Hicks, A. (2023). Base Fee Manipulation In Ethereum's EIP-1559 Transaction Fee Mechanism. *arXiv (Cornell University).* Advance online publication. https://doi.org/10.48550/arXiv.2304.11478

Barron, J. (n.d.). *The Importance of Account Management.* Utility Team. https://www.utilityteam.co.uk/blog/the-importance-of-account-management

Basel Committee on Banking Supervision (BCBS). (2014). *Guidelines Sound management of risks related to money laundering and financing of terrorism (BCBS Guideline)*. Updated July 2020. https://www.bis.org/bcbs/publ/d505.pdf

Basel Committee on Banking Supervision (BCBS). (2015). *Consultative Document General guide to account opening*. https://www.bis.org/bcbs/publ/d331.pdf

Bharadwaj, C. (2023, March 3). *Custodial vs. non-custodial wallets: understanding the difference points.* Appinventiv. https://appinventiv.com/blog/custodial-vs-non-custodial-wallets/

Bhatt, M. (n.d.). Real-Time Payments: What Are They and How Do They Work? *Host Merchant Services*. https://www.hostmerchantservices.com/articles/real-time-payments-what-are-they-and-how-do-they-work/

Biryukov, A., Khovratovich, D., & Tikhomirov, S. (2018). Privacy-preserving KYC on Ethereum. *European Society for Socially Embedded Technologies (EUSSET).* Advance online publication. https://doi.org/10.18420/blockchain2018_09

Blocktrade. (2021). *The keys to crypto kingdom: wallet address, public and private keys explained*. https://blocktrade.com/wallet-addresses-public-and-private-keys-explained/

Böhl, L. (2022, January 10). Unterschied zwischen Coin und Token erklärt [Difference between Coin and Token explained]. *Stuttgarter Nachrichten*. https://www.stuttgarter-nachrichten.de/inhalt.unterschied-coin-token-mhsd.bd0fe6d7-5a3f-40b9-b9bf-847c44002047.html

Bots. (2022). *Crypto Coin Vs. Token: Understanding the Difference*. https://www.bots.io/news/crypto-coin-vs-token-difference

Brennen, A. (2021, May 28). *10 Best Practices for Secure Online Payment Processing.* Rapyd. https://www.rapyd.net/blog/secure-online-payment-processing/

Bruhn, M., & Georgi, D. (2006). *Dienstleistungsmanagement in Banken [Service management in banks]*. Bankakademie Verlag GmbH.

Camenisch, J., Kohlweiss, M., & Soriente, C. (2009). An Accumulator Based on Bilinear Maps and Efficient Revocation for Anonymous Credentials. In S. Jarecki & G. Tsudik (Eds.), *Lecture Notes in Computer Science: Vol. 5443. Public Key Cryptography - PKC 2009: 12th International Conference on Practice and Theory in Public Key Cryptography, Irvine, CA, USA, March 18-20, 2009. Proceedings* (Vol. 5443, pp. 481–500). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-00468-1_27

Chainalysis. (2022). *The 2022 Crypto Crime Report*. https://go.chainalysis.com/rs/503-FAP-074/images/Crypto-Crime-Report-2022.pdf

Chitimira, H., & Munedzi, S. (2022). Overview international best practices on customer due diligence and related anti-money laundering measures. *Journal of Money Laundering Control*, *26*(7), 53–62. https://doi.org/10.1108/JMLC-07-2022-0102

Crowson, A. (2023, May 20). *How to set up a basic cryptocurrency wallet.* Crypto Vantage. https://www.cryptovantage.com/guides/setting-up-crypto-wallet/

Crypto.com. (2022, April 26). *What Is a Crypto Wallet? A Beginner's Guide*. https://crypto.com/university/crypto-wallets

Crypto.com. (2023, February 17). *Custodial vs Non-Custodial Wallets*. https://crypto.com/university/custodial-vs-non-custodial-wallets#:~:text=Key%20Takeaways%3A,and%2C%20therefore%2C%20their%20funds.

Deloitte. (n.d.). *Modernizing transaction banking*. https://www2.deloitte.com/us/en/pages/financial-services/articles/modernizing-transaction-banking-technology-service-externalization.html

DeMarco, J. (2022, June 16). *What's the Difference Between a Certified Check and a Cashier's Check?* SoFi Learn. https://www.sofi.com/learn/content/certified-check-vs-cashiers-check/

Derleder, P., Knops, K.-O., & Bamberger, H. G. (2017a). *Deutsches und europäisches Bank- und Kapitalmarktrecht – Band 1 [German and European Banking and Capital Markets Law – Volume 1]*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-52807-5

Derleder, P., Knops, K.-O., & Bamberger, H. G. (2017b). *Deutsches und europäisches Bank- und Kapitalmarktrecht – Band 2 [German and European Banking and Capital Markets Law – Volume 2]*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-52805-1

Di Pierro, M. (2017). What Is the Blockchain? *Computing in Science & Engineering*, *19*(5), 92–95. https://doi.org/10.1109/MCSE.2017.3421554

Donmez, A., & Karaivanov, A. (2022). Transaction fee economics in the Ethereum blockchain. *Economic Inquiry*, *60*(1), 265–292. https://doi.org/10.1111/ecin.13025

Eidgenössische Finanzmarktaufsicht [*Swiss Financial Market Supervisory Authority*] (FINMA). (2023). *Aufsicht im Bankenbereich [Banking supervision]*. https://www.finma.ch/de/ueberwachung/banken-und-wertpapierhaeuser/

Ethereum. (2023, May 31). *ERC-20 Token Standard*. https://ethereum.org/en/developers/docs/standards/tokens/erc-20/

European Central Bank (ECB). (2020, August 12). *What are TARGET2 balances?* https://www.ecb.europa.eu/ecb/educational/explainers/tell-me-more/html/target2_balances.en.html

European Central Bank (ECB). (2023). *What is TARGET2?* https://www.ecb.europa.eu/paym/target/target2/html/index.en.html

European Payments Council. (2021, October 7). *Digitalisation trends in the Swiss payment landscape*. https://www.europeanpaymentscouncil.eu/news-insights/insight/digitalisation-trends-swiss-payment-landscape

Expatica. (2023, May 8). *Swiss money transfers: how to send money abroad*. https://www.expatica.com/ch/finance/money-management/swiss-money-transfer-212632/

Farley, M. (2018, November 13). *Best Practices for Bank Account Management.* Coalition Greenwich. https://www.greenwich.com/corporate-banking/best-practices-bank-account-management

Federal Act on Combating Money Laundering and Terrorist Financing, Anti-Money Laundering Act (AMLA), classified compilation of the federal law 955.0 (1997). https://www.fedlex.admin.ch/eli/cc/1998/892_892_892/en

Financial Action Task Force (FATF). (2023). *International Standards on Combating Money Laundering and the Financing of Terrorism & Proliferation*. https://www.fatf-gafi.org/content/dam/fatf-gafi/recommendations/FATF%20Recommendations%202012.pdf.coredownload.inline.pdf

Gemini. (2022, June 24). *Custody Fee Schedule*. https://www.gemini.com/fees/custody-fee-schedule#section-notice-of-changes

Hayes, A. (2023, April 23). *Blockchain Facts: What Is It, How It Works, and How It Can Be Used*. Investopedia. https://www.investopedia.com/terms/b/blockchain.asp#:~:text=While%20the%20hackers%20may%20have,as%20most%20others)%20are%20encrypted.

Heires, K. (2016). The risks and rewards of blockchain technology. *Risk Management*, *63*(2). https://go.gale.com/ps/i.do?id=GALE%7CA446004226&sid=googleScholar&v=2.1&it=r&linkaccess=abs&issn=00355593&p=AONE&sw=w&userGroupName=anon%7E9c7b6196&aty=open+web+entry

Heller, D., Nellen, T., & Sturm, A. (2000). 2000, The Swiss Interbank Clearing System. *ResearchGate*. https://www.researchgate.net/profile/Thomas-Nellen/publication/228595529_2000_The_Swiss_Interbank_Clearing_System/links/5404438f0cf2bba34c1c5c40/2000-The-Swiss-Interbank-Clearing-System.pdf

Hewa, T., Ylianttila, M., & Liyanage, M. (2021). Survey on blockchain based smart contracts: Applications, opportunities and challenges. *Journal of Network and Computer Applications*, *177*, Article 102857. https://doi.org/10.1016/j.jnca.2020.102857

International Monetary Fund. (2023). *GDP, current prices*. https://www.imf.org/external/datamapper/NGDPD@WEO/OEMDC/ADVEC/WEOWORLD

Jaag, C., & Bach, C. (2015). Chapter 6 - The Effect of Payment Reversibility on E-commerce and Postal Quality. In D. K. C. Lee (Ed.), *Handbook of digital currency: Bitcoin, innovation, financial instruments, and big data* (1st ed., pp. 139–151). Elsevier Academic Press. https://doi.org/10.1016/B978-0-12-802117-0.00006-0

Kaufman, G. G. (Ed.). (1992). *Banking Structures in Major Countries* (Vol. 6). Springer Netherlands. https://doi.org/10.1007/978-94-011-2946-6

Kaulartz, M., & Heckmann, J. (2016). Smart Contracts – Anwendungen der Blockchain-Technologie [Smart contracts - applications of blockchain technology]. *Computer und Recht*, *32*(9), 618-624. https://doi.org/10.9785/cr-2016-0923

Kowsmann, P., & Talley, I. (2022, March 1). What Is Swift and Why Is It Being Used to Sanction Russia?. *The Wall Street Journal*, https://www.wsj.com/articles/swift-banking-system-sanctions-biden-11645745909

Landtwing-Leupi M. (2020, July 2). *GwG Compliance für virtuelle Assets in der Schweiz [AMLA Compliance for Virtual Assets in Switzerland].* MME Compliance AG. https://www.mme.ch/de-ch/magazin/artikel/gwg-compliance-fuer-virtuelle-assets-in-der-schweiz

Lessambo, F. I. (2023). *Anti-Money Laundering, Counter Financing Terrorism and Cybersecurity in the Banking Industry*. Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-23484-2

LinkedIn. (n.d.). *How does blockchain compare to other payment methods in terms of speed, cost, and security?.* LinkedIn. https://www.linkedin.com/advice/0/how-does-blockchain-compare-other-payment-methods

Malhotra, D., Saini, P., & Singh, A. K. (2022). How Blockchain Can Automate KYC: Systematic Review. *Wireless Personal Communications*, *122*(2), 1987–2021. https://doi.org/10.1007/s11277-021-08977-0

Marsh, W. B., & Maniff, J. L. (2017). *Banking on Distributed Ledger Technology: Can It Help Banks Address Financial Inclusion?*. Federal Reserve Bank of Kansas City. https://doi.org/10.18651/er/3q17maniffmarsh

McKinsey. (n.d.). *Building a successful payments system*. McKinsey&Company. https://www.mckinsey.com/industries/financial-services/our-insights/banking-matters/building-a-successful-payments-system

Mekuli, A. (2022, November 10). *Banking in Switzerland: How To Open a Swiss Bank Account [2022].* Studying in Switzerland. https://studyinginswitzerland.com/swiss-bank-accounts/#

Mulhim, H. (2022, September 21). *Opinion: Why crypto businesses need anti-money laundering regulations.* World Economic Forum. https://www.weforum.org/agenda/2022/09/why-crypto-businesses-need-anti-money-laundering-regulations/

Nahar, P. (2022, January 17). Crypto class: Difference between crypto coin & token. *Economic Times*. https://economictimes.indiatimes.com/markets/cryptocurrency/crypto-class-difference-between-crypto-coin-token/articleshow/88947666.cms?from=mdr

Oad, A., Razaque, A., Tolemyssov, A., Alotaibi, M., Alotaibi, B., & Zhao, C. (2021). Blockchain-Enabled Transaction Scanning Method for Money Laundering Detection. *Electronics*, *10*(15), 1766. https://doi.org/10.3390/electronics10151766

Ordinance of the Swiss Financial Market Supervisory Authority on combating money laundering and terrorist financing in the terrorist financing in the financial sector, FINMA Anti-Money Laundering Ordinance (AMLO-FINMA), classified compilation of the federal law 955.033.0 (2015). https://www.fedlex.admin.ch/eli/cc/2015/390/de

Page, M. J., Moher, D., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T [Tianjing], Loder, E. W., Mayo-Wilson, E., McDonald, S., McKenzie, J. E. (2021). Prisma 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ (Clinical Research Ed.)*, *372*, n160. https://doi.org/10.1136/bmj.n160

Parra-Moyano, J., & Ross, O. (2017). KYC Optimization Using Distributed Ledger Technology. *Business & Information Systems Engineering*, *59*(6), 411–423. https://doi.org/10.1007/s12599-017-0504-2

Peters, G. W., & Panayi, E. (2016). Understanding Modern Banking Ledgers Through Blockchain Technologies: Future of Transaction Processing and Smart Contracts on the Internet of Money. In P. Tasca, T. Aste, L. Pelizzon, & N. Perony (Eds.), *New Economic Windows. Banking Beyond Banks and Money: A Guide to Banking Services in the Twenty-First Century* (1st ed., pp. 239–278). Springer International Publishing. https://doi.org/10.1007/978-3-319-42448-4_13

Peyton, A. (2016, April 18). *Switzerland's new RTGS system paves the way for ISO 20022.* Fintech Futures. https://www.fintechfutures.com/2016/04/switzerlands-new-rtgs-system-paves-the-way-for-iso-20022/

Pratz, A., Lankinen, L., Weiss, M., & Wegner, T. (2021). *Capturing the value in 'knowing your customer', while cutting costs and risks*. Strategy&. https://www.strategyand.pwc.com/ch/en/industries/financial-services/kyc.html

PXL Vision. (n.d.). *Krypto ohne KYC-Prüfung kaufen [Buy crypto without KYC verification]*. PXL Vision. https://www.pxl-vision.com/de/kyc/krypto-ohne-kyc

Qiu, T., Zhang, R., & Gao, Y. (2019). Ripple vs. SWIFT: Transforming Cross Border Remittance Using Blockchain Technology. *Procedia Computer Science*, *147*, 428–434. https://doi.org/10.1016/j.procs.2019.01.260

Raskin, M., & Yermack, D. (Eds.). (2018). *Digital currencies, decentralized ledgers and the future of central banking*. Edward Elgar Publishing. https://doi.org/10.4337/9781784719227.00028

Ratnawat, N., Pandey, S., Paradkar, R., & Banerjee, S. (2022). Optimizing the KYC Process using a Blockchain based approach. *ITM Web of Conferences*, *44*, 03039. https://doi.org/10.1051/itmconf/20224403039

Roughgarden, T. (2020). Transaction Fee Mechanism Design for the Ethereum Blockchain: An Economic Analysis of EIP-1559. *ResearchGate*. https://www.researchgate.net/publication/346578431_Transaction_Fee_Mechanism_Design_for_the_Ethereum_Blockchain_An_Economic_Analysis_of_EIP-1559

Schwarz, M. (2023, June 7). *Bitcoin Wallet Vergleich: alles, was man über Bitcoin Wallets wissen muss & beste Wallets 2023 [Bitcoin wallet comparison: everything you need to know about bitcoin wallets & best wallets 2023].* coincierge. https://coincierge.de/wallets/#Bitcoin_Wallet_Gebuehren

Scott, S. V., & Zachariadis, M. (2013). *The Society for Worldwide Interbank Financial Telecommunication (SWIFT)*. Taylor & Francis. https://doi.org/10.4324/9781315849324

Seth, S. (2023, April 26). *What Is the SWIFT Banking System?.* Investopedia. https://www.investopedia.com/articles/personal-finance/050515/how-swift-system-works.asp

Simmons, G. J. (1979). Symmetric and Asymmetric Encryption. *ACM Computing Surveys*, *11*(4), 305–330. https://doi.org/10.1145/356789.356793

Singh, K., & Best, P. (2019). Anti-Money Laundering: Using data visualization to identify suspicious activity. *International Journal of Accounting Information Systems*, *34*, Article 100418. https://doi.org/10.1016/j.accinf.2019.06.001

Singhal, N., Sharma, M. K., Samant, S. S., Goswami, P., & Reddy, Y. A. (2020). Smart KYC Using Blockchain and IPFS. In V. K. Gunjan, S. Senatore, A. Kumar, X. Z. Gao, & S. Merugu (Eds.), *Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies* (Vol. 643, pp. 77–84). Springer Singapore. https://doi.org/10.1007/978-981-15-3125-5_9

Soltani, R., Trang Nguyen, U., & An, A. (2018). A New Approach to Client Onboarding Using Self-Sovereign Identity and Distributed Ledger. *IEEE Conference Publication*, 1129–1136. https://doi.org/10.1109/Cybermatics_2018.2018.00205

Sullivan, K. (2015). Know Your Customer and Customer Identification Program. In K. Sullivan (Ed.), *Anti–Money Laundering in a Nutshell* (pp. 69–100). Apress. https://doi.org/10.1007/978-1-4302-6161-2_5

Suratkar, S., Shirole, M., & Bhirud, S. (2020). Cryptocurrency Wallet: A Review. *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP).* IEEE. https://doi.org/10.1109/icccsp49186.2020.9315193

Swiss Bankers Association (SBA). (n.d.). *Combating money laundering*. https://www.swissbanking.ch/en/topics/regulation-and-compliance/the-fight-against-money-laundering

Swiss Bankers Association (SBA). (2023). *Information for bank clients*. https://www.swissbanking.ch/en/financial-centre/information-for-bank-clients-and-companies/information-for-bank-clients

Swiss Criminal Code, (SCC), classified compilation of the federal law 311.0 (1937). https://www.fedlex.admin.ch/eli/cc/54/757_781_799/en

Swiss Euro Clearing Bank. (n.d.). *The bank*. https://www.secb.de/en/

Swiss Federal Court, BGE 124 III 313 pp. 316 – 317 (1998). https://www.bger.ch/ext/eurospider/live/de/php/clir/http/index.php?highlight_docid=atf%3A%2F%2F124-III-313%3Ade&lang=de&zoom=&type=show_document

Swiss Infrastructure and Exchange (SIX). (2022, July 28). *How and When Will Swiss Instant Payments Work "Instantly"?*. https://www.six-group.com/en/blog/2022/swiss-instant-payments.html

Swiss Infrastructure and Exchange (SIX). (2023a, April 15). *Payments to the Eurozone.* https://www.six-group.com/en/products-services/banking-services/interbank-clearing/eurosic/payments-euro-zone.html

Swiss Infrastructure and Exchange (SIX). (2023b, April 15). *Euro Transactions to Switzerland and Liechtenstein*. https://www.six-group.com/en/products-services/banking-services/interbank-clearing/eurosic/payments-ch-li.html

Swiss Infrastructure and Exchange (SIX). (2023c, April 15). *SIC System*. https://www.six-group.com/en/products-services/banking-services/interbank-clearing/sic.html

Swiss Infrastructure and Exchange (SIX). (2023d, May 7). *euroSIC System*. https://www.six-group.com/en/products-services/banking-services/interbank-clearing/eurosic.html

Swiss Infrastructure and Exchange (SIX). (2023e, June 25). *Zahlungssystem SIC [SIC payment system]*. https://www.six-group.com/de/products-services/banking-services/interbank-clearing/sic.html

Swiss National Bank. (2023, May 3). *Zahlungsverkehr [Payment transactions]*. https://www.snb.ch/de/iabout/paytrans

Swissquote. (2023). *Pricing*. https://en.swissquote.com/crypto-assets/pricing

Thomson Reuters. (2016). *Thomson Reuters 2016 Know Your Customer Surveys Reveal Escalating Costs and Complexity*. https://www.thomsonreuters.com/en/press-releases/2016/may/thomson-reuters-2016-know-your-customer-surveys.html

Thomson Reuters. (2022). *2022 Thomson Reuters Anti-Money Laundering Insights Survey*. https://legal.thomsonreuters.com/en/insights/reports/2022-thomson-reuters-anti-money-laundering-insights-survey/form?gatedContent=%252Fcontent%252Fewp-marketing-websites%252Flegal%252Fgl%252Fen%252Finsights%252Freports%252F2022-thomson-reuters-anti-money-laundering-insights-survey

Trojanvilla.com. (2022, December 26). *MT103*. https://trojanvilla.com/mt103/

Union Bank of Switzerland (UBS). (2023a). *Investment fund account*. https://www.ubs.com/ch/en/private/accounts-and-cards/accounts/fund-account.html

Union Bank of Switzerland (UBS). (2023b). *Savings account*. https://www.ubs.com/ch/en/private/accounts-and-cards/accounts/savings-account.html

Union Bank of Switzerland (UBS). (2023c, May 7). *Fees for payments*. https://www.ubs.com/ch/en/help/payments/prices.html

United Nations. (2023, July 20). *Money Laundering*. https://www.unodc.org/unodc/en/money-laundering/overview.html

Validatis. (2023, April 12). *KYC: Was ist »Know Your Customer«? [What is »Know Your Customer«?].* https://www.validatis.de/kyc-prozess/news-fachwissen/kyc-know-your-customer/

Vertesi, O. (2022). *ERC20 token and smart contract programming for a blockchain start-up*. https://intranet.blockchainpresence.net/developement-thesis/

Vogelsteller, F., & Buterin, V. (2015, November 19). *ERC-20: Token Standard*. https://eips.ethereum.org/EIPS/eip-20

Wandelt, M., & Werner, A. (2020). *Digital Onboarding and KYC Report 2020: Fighting Financial Crime with Retech.* Deloitte. https://thepaypers.com/reports/download/digital-onboarding-and-kyc-report2020?cid=f5abab5fcb6fa78c640023bfa33b7425&utm_source=Maileon&utm_medium=email&utm_campaign=The+Paypers+-+Download+Confirmation+Report+Trigger+Mail+4.2&utm_content=https%3A%2F%2Fthepaypers.com%2Freports%2Fdownload%2Fdigital-onboarding-and-kyc-report-2020%3Fcid%3Df5abab5fcb6fa78c640023bfa33b7425

Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., & Wang, F. (2018). An Overview of Smart Contract: Architecture, Applications, and Future Trends. *IEEE Intelligent Vehicles Symposium*, 108–113. https://doi.org/10.1109/IVS.2018.8500488

Wheeler, J. (2021, October 19). *What Does CDD (Customer Due Diligence) Mean for Banks and Financial Companies?*. Jumio. https://www.jumio.com/cdd-customer-due-diligence/

Wheeler, J. (2022, September 11). *The Relationship Between Know Your Customer (KYC) & Customer Due Diligence (CDD).* Jumio. https://www.jumio.com/kyc-vs-cdd/

Willi, R. (2021). *Solidity coding: Implementing an equity token and further use cases*. https://intranet.blockchainpresence.net/developement-thesis/

Wouters, S. (2021, April 28). *Digital Asset Custody in 2021.* Blockdata. https://www.blockdata.tech/blog/general/digital-asset-custody-in-2021

Xu, C., Liu, C., Nie, D., & Gai, L. (2021). How Can a Blockchain-Based Anti-Money Laundering System Improve Customer Due Diligence Process?. *Journal of Forensic and Investigative Accounting*, *13*(2). https://scholar.google.com/citations?view_op=view_citation&hl=en&user=NkX3TxIAAAAJ&citation_for_view=NkX3TxIAAAAJ:u5HHmVD_uO8C

Yadav, P., & Chandak, R. (2019). Transforming the Know Your Customer (KYC) Process using Blockchain. *2019 International Conference on Advances in Computing, Communication and Control (ICAC3). IEEE.* https://doi.org/10.1109/ICAC347590.2019.9036811

Zanzi, A. (2022, August 25). *Marktbeobachtung zu den Gebühren von Schweizer Bankkonten [Market observation on Swiss bank account fees].* Eidgenössisches Departement für Wirtschaft, Bildung und Forschung (DEFR). https://www.preisueberwacher.admin.ch/dam/pue/de/dokumente/studien/rapport_frais_bancaires_2022.pdf.download.pdf/Marktbeobachtung%20zu%20den%20Geb%C3%BChren%20von%20Schweizer%20Bankkonten%202022.pdf

Zürcher Kantonalbank. (2023, April 15). *Wie erfasse ich eine Zahlung im eBanking? [How do I enter a payment in eBanking?]*. https://www.zkb.ch/de/hilfe/skf/zahlung-im-ebanking.html

Appendix

Appendix A: Smart contract testing guide

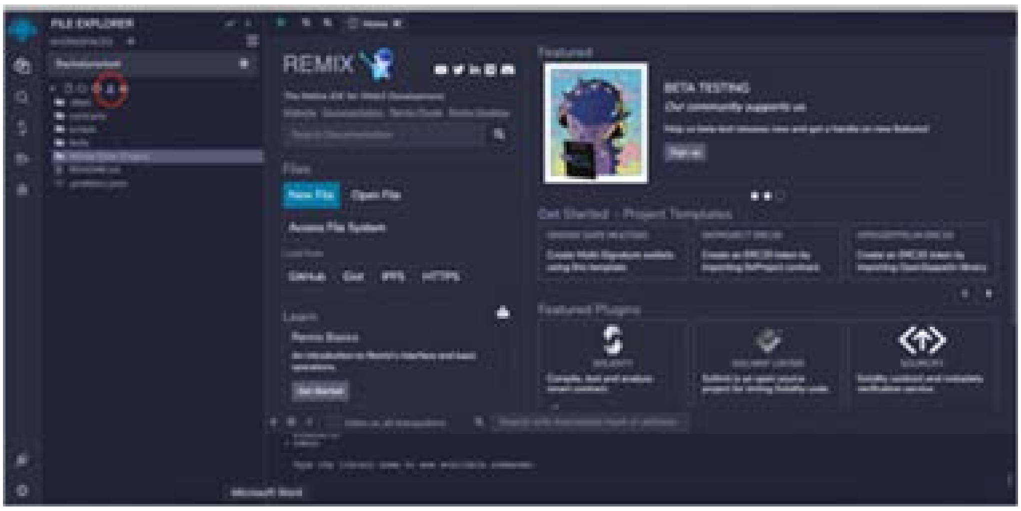
Smart contract set up:

1. Go to Remix IDE: https://remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js

2. Upload BCP_informed, the fake oracle and the KYCrypto contract.

Figure A 1

Uploading Contracts

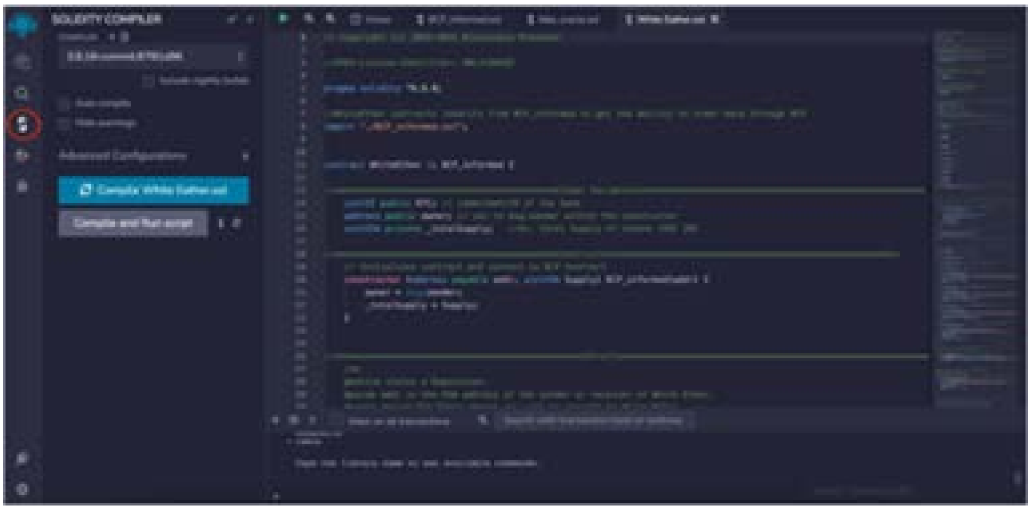


Note. Own Illustration.

3. Compile the fake oracle and the KYCrypto Contract.

Figure A 2

Compiling contracts



Note. Own Illustration.

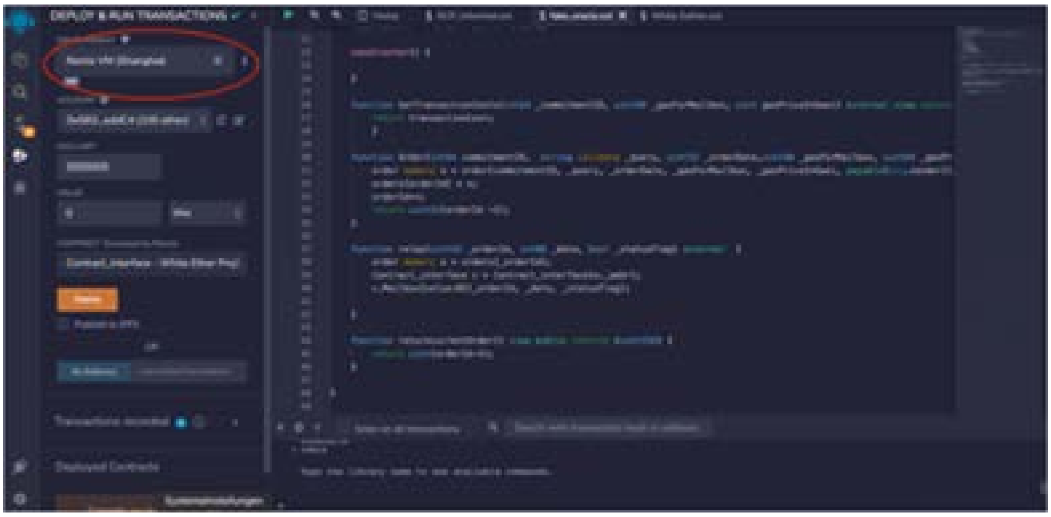
When compiling the fake oracle, the compiler will give out warnings, just ignore these. As long as the warnings are orange in color, the contract will still work.
If there is a red error the code will not work, if that happens check if all the versions of the code are up to date. If that is the case, there is a problem with the code, that needs to be addressed.

4. After compiling the contracts, we are now ready for deployment. Go to the Deploy and Run Transactions tab (under the compiling tab).

5. Choose a Remix VM Environment. Do not use Injected Provider-Metamask or any other injected providers.

Figure A 3

Choosing an environment

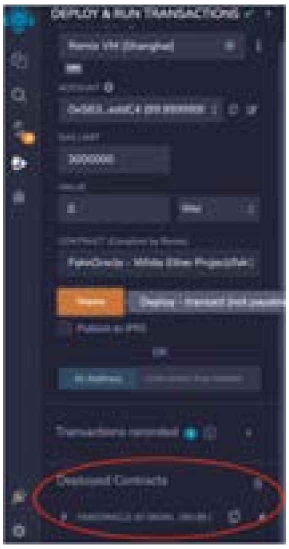


Note. Own Illustration.

6. Go to the fake oracle tab.
   a. Under CONTRACT, choose the FakeOracle Contract. Not the interface.
   b. Press deploy.
   c. If the deployment works you should be able to see the deployed contract under Deployed Contracts. The contract is now deployed.

Figure A 4

Fake oracle deployment

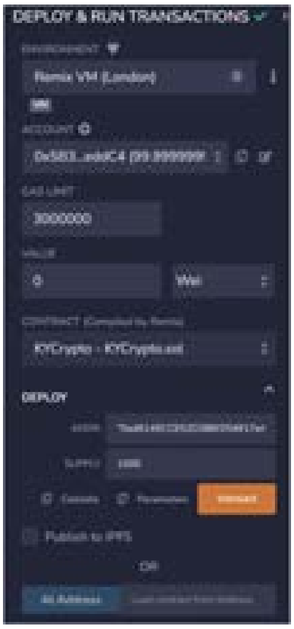


Note. Own Illustration.

7. Go to the KYCrypto tab
   a. Choose the KYCrypto Contract under CONTRACT.
   b. Before the deployment of this contract, 2 things have to be specified:
      i. Address: Here you have to put in the address of the fake oracle we just deployed. Go to the deployed contract and copy and paste it.
      ii. Supply: The total supply (in the deploy sub-section) of our token, put in any large number you like. If the number is too small you will run into problems while testing.
   c. Press deploy.

Figure A 5

KYCrypto deployment



Note. Own Illustration.

Registering an account:

1. In the KYCrypto smart contract, call the *_Account_Registration* function. To call the function a number must be put in for the bankId and a value must be sent with the transaction.

2. Use the relay function in the fake oracle. To use this function three values need to be put in:
    a. *_orderId*: Number of the order that should be processed. The orders start at 0 and then go up. If the current order number is not clear, the *returncurrentOrder* function can give out the number of the current order.
    b. *_data*: This value gives the actual information, in this version of the code it is simply checked if the customer is on the whitelist or not.
    To whitelist a person the number that is put in needs to be “1”.

   c. *_statusFlag*: This value is there to verify that the requested information is present. The value needs to be “1”.

3. The Account is now registered and can use the different functions in the smart contract.

Appendix B: Smart contract code

```
// Copyright (c) 2019-2023 Smart Contracts Lab

//SPDX-License-Identifier: UNLICENSED

pragma solidity ^0.8.0;

// KYCrypto contracts inherits BCP_informed contract to order data through BCP
import "./bcp_informed.sol";

contract KYCrypto is BCP_informed {


//--------------------------Global_Variables-------------------------

    // set to msg.sender within the constructor
    address public owner;

    // total Supply of KYCrypto (ERC20 Token)
    uint256 private _totalSupply;


//----------------------------Constructor----------------------------

    // Initializes contract and connect to the BCP Contract
    constructor (address payable addr, uint256 Supply) BCP_informed(addr) {
        owner = msg.sender;
        _totalSupply = Supply;
    }


//------------------------------Structs------------------------------

    /**
    @notice stores a deposition.
    @param address (addr) is the Externally Owned Account (EOA) address of the
sender or receiver of KYCrypto.
    @param amount of cryptocurrency you want to convert into KYCrypto.
    */
    struct Deposit{
        address payable addr;
        uint amount;
    }


    /**
    @notice stores information of a whitelisted Account holder.
    @param balance is the amount of KYCrypto of a user.
    @param status signifies if user is whitelisted.
    */
    enum Status  {NONE, BLOCKED, APPROVED, DECLINED}

    struct User{
        uint balance;
        Status status;
    }


//----------------------------Mappings-------------------------------

    // maps uint to Deposit struct
    mapping(uint => Deposit) public deposits;

    // maps address to User struct
    mapping(address => User) public users;

    // maps address to uint
    mapping(address => uint) public declinedusers;

    // maps the token owners address to the address of the spender and the token
amount allowed
    mapping(address => mapping(address => uint256)) private _allowed;


 //------------------------------Events------------------------------

        event regist(
        address payable adr,
        uint _value,
        uint _id
    );

        event transf(
        address from,
        address to,
        uint amount
    );

        event pullKYCrypto(
        address from,
        uint amount
    );

        event pulldeclineKYCrypto(
        address from
    );

        event blockedAcount(
        address from
    );

        event unblockedAcount(
        address from
    );

        event unusedKYCrypto(
        address from,
        uint amount
    );

        event Approval(
        address from,
        address to,
        uint256 amount
    );


//----------------------------modifier----------------------------
    modifier onlyOwner() {
        require(msg.sender==owner, "Sender not authorized.");
        _;
    }


    function initialofall() public onlyOwner{
        users[owner].status = Status.APPROVED;
    }


//----------------------------Helper_function----------------------------

    /**
    @dev this function is needed for the inquire function. The function converts
a ledger address to an ASCII string.
    */
    function addresstoAsciiString(address x) internal pure returns (string
memory) {
        bytes memory s = new bytes(42);
        bytes memory name = abi.encodePacked("0x");

        for (uint i = 0; i < 20; i++) {
             bytes1 b = bytes1(uint8(uint(uint160(x)) / (2**(8*(19 - i)))));
             bytes1 hi = bytes1(uint8(b) / 16);
             bytes1 lo = bytes1(uint8(b) - 16 * uint8(hi));
             s[2*i] = char(hi);
             s[2*i+1] = char(lo);
        }
        name = abi.encodePacked(name, s);
        return string (name);
        }

    /**
    @dev this function is needed for the addresstoAsciiString function.
    */
    function char(bytes1 b) internal pure returns (bytes1 c) {
        if (uint8(b) < 10) return bytes1(uint8(b) + 0x30);
        else return bytes1(uint8(b) + 0x57);
        }


//----------------------------Functions----------------------------

    // provides the total token supply information
    function totalSupply() view public returns (uint256) {
        return _totalSupply;
    }


    // provides balance of an account
    function balanceOf(address _addr) public view returns (uint256) {
        return users[_addr].balance;
    }


    // enables the contract owner to increase token supply
    function Token_mint(uint256 amount) public onlyOwner returns (bool success)
{
    _totalSupply += amount;
    return true;
    }


    // enables the contract owner to decrease token supply
    function Token_burn(uint256 amount) public onlyOwner returns (bool success)
{
    _totalSupply -= amount;
    return true;
    }


    // returns the approved number of coins that can be spent from a certain
account to another
    function allowance(address _from, address _to) view public returns(uint256)
{
        return _allowed[_from][_to];
    }


    // enables a spender to withdraw a set number of tokens from a specified
account
    function approve(address _to, uint256 _amount) public returns (bool) {
        require(_to != address(0));

        User storage from = users[msg.sender];
        User storage to = users[_to];
        require(from.status == Status.APPROVED, "To approve the spending of
KYCrypto you need to have a whitelisted account");
        require(to.status == Status.APPROVED, "To get approved to spend
KYCrypto you need to have a whitelisted account");

        _allowed[msg.sender][_to] = _amount;
        emit Approval(msg.sender, _to, _amount);
        return true;
    }


    // enables a spender to increase the token approval from a specified account
    function increaseApproval(address _to, uint256 addedApproval) public re-
turns (bool) {
        require(_to != address(0));

        User storage from = users[msg.sender];
        User storage to = users[_to];
        require(from.status == Status.APPROVED, "To approve KYCrypto you need
to have a whitelisted account");
        require(to.status == Status.APPROVED, "To get approved to spend
KYCrypto you need to have a whitelisted account");

        _allowed[msg.sender][_to] += addedApproval;
        emit Approval(msg.sender, _to, _allowed[msg.sender][_to]);
        return true;
    }


    // enables a spender to decrease the token approval from a specified account
    function decreaseApproval(address _to, uint256 subtractedApproval) public
returns (bool) {
        require(_to != address(0));

        User storage from = users[msg.sender];
        User storage to = users[_to];
        require(from.status == Status.APPROVED, "To approve KYCrypto you need
to have a whitelisted account");
        require(to.status == Status.APPROVED, "To get approved to spend
KYCrypto you need to have a whitelisted account");

        _allowed[msg.sender][_to] -= subtractedApproval;
        emit Approval(msg.sender, _to, _allowed[msg.sender][_to]);
        return true;
    }


    // executes transfers of a specified number of tokens from a specified
address
    function transferFrom(address _from, address _to, uint256 _amount) public
returns (bool success) {
        User storage from = users[_from];
        User storage to = users[_to];
        User storage beneficiary = users[msg.sender];

        require(from.balance >= _amount, "Insufficient KYCrypto balance.");
        require(_allowed[_from][msg.sender] >= _amount, "Transfer value exceeds
Allowance.");
        require(from.status == Status.APPROVED, "The sender of KYCrypto has to
have a whitelisted account");
        require(to.status == Status.APPROVED, "The receiver of KYCrypto has to
have a whitelisted account");
        require(beneficiary.status == Status.APPROVED, "You need a whitelisted
Account to spend KYCrypto");
        from.balance -= _amount;
        to.balance += _amount;
        _allowed[_from][msg.sender] -= _amount;

        emit transf(_from, _to, _amount);
        return true;
    }


    /**
    @dev function transfer transfers KYCrypto to another whitelisted Account
    @param _to address to which KYCrypto is sent
    @param _amount amount of KYCrypto sent
    */
    function transfer(address _to, uint _amount) public returns (bool success){
        User storage from = users[msg.sender];
        User storage to = users[_to];
        require (from.status == Status.APPROVED, "To send KYCrypto you need to have a whitelisted Account");
        require(from.balance >= _amount, "Insufficient KYCrypto balance.");
        require (to.status == Status.APPROVED, "The receiver must be a whitelisted Account");
        from.balance -= _amount;
        to.balance += _amount;
        emit transf(msg.sender, _to, _amount);
        return true;
        }


    /**
    @dev function _Account_Registration is used to if a person wants to exchange cryptocurrency into KYCrypto
    @param _bankid the commitmentID of the bank where the user has his money
    */
    function _Account_Registration(uint32 _bankid) external payable returns (bool success) {
        require(users[msg.sender].status == Status.NONE, "Account is frozen or already registered");
        require(msg.value > 0, "A value must be sent with the function call!");
        address payable addresstoRegsiter = payable(msg.sender);
        uint32 _commitmentID = _bankid;
        uint32 _gasForMailbox = 200000;    //Our Mailbox function uses at most 200'000 gas
        uint gasPriceInGwei = 30;
        uint transactionCost = BCP.GetTransactionCosts(int64(uint64(_commitmentID)), _gasForMailbox,gasPriceInGwei);
        require(msg.value >= transactionCost, "Value of the Transaction too low.");
        uint orderID = BCP.Order{value:transactionCost}(int64(uint64(_commitmentID)),addresstoAsciiString (addresstoRegsiter),uint32(block.timestamp),20000,uint64(gasPriceInGwei));
        uint amount_sent = msg.value - transactionCost;
        _totalSupply -= amount_sent;
        deposits[orderID] = Deposit(addresstoRegsiter, amount_sent);
        emit regist(payable(msg.sender), msg.value, _bankid);
        return true;
        }

    function _depositKYCrypto() public payable returns (bool success) {
        require(users[msg.sender].status == Status.APPROVED, "Account is blocked or not yet registered");
        require(msg.value > 0, "A value must be sent with the function call!");
        uint put_amount = (99 * msg.value)/100;
        uint put_fee = (1 * msg.value)/100;
        users[owner].balance += put_fee;
        _totalSupply -= msg.value;
        users[msg.sender].balance += put_amount;
        return true;

    }


    /**
    @dev function Mailbox is used to reiceive data from BCP
    @param _orderID uint that identifies a specific order (is constant)
    @param _data is the finally requested information behind the order
    @param _statusFlag is a control variable that shows if the incoming trans-
action contains the datapoint
    */
    function Mailbox(uint32 _orderID, int88 _data, bool _statusFlag) external
payable override onlyBCP {

        Deposit memory t = deposits[_orderID];
        uint amount = t.amount;
        uint cryptodeposit = (99* amount)/100;
        uint fee_for_registering = (1 * amount)/100;
        if(_statusFlag && _data == 1) {
            users[t.addr].balance += cryptodeposit;
            users[owner].balance += fee_for_registering;
            users[t.addr].status=Status.APPROVED;
        }
        else {
            declinedusers[t.addr] = amount;
            users[t.addr].status = Status.DECLINED;
        }
        delete deposits[_orderID];

        }


    /**
    @dev function _withdrawthedeclinedCrypto will be called by persons whose
cryptocurrency is declined to get it back.
    */
    function _withdrawthedeclinedCrypto() public payable returns (bool success)
{
        uint moneyback = declinedusers[msg.sender];
        delete declinedusers[msg.sender];
        payable(msg.sender).transfer(moneyback);
        emit pulldeclineKYCrypto(msg.sender);

        if(users[msg.sender].status == Status.DECLINED) {
            users[msg.sender].status = Status.NONE;
        }
        return true;
    }


    /**
    @dev function _withdrawKYCrypto with this function KYCrypto can be exchanged
to Ether
    @param _pull_amount the amount of White Ether which should be exchanged
back
    **/
    function _withdrawKYCrypto(uint _pull_amount) public payable returns (bool
success) {
        require(users[msg.sender].status == Status.APPROVED);
        require(_pull_amount <= users[msg.sender].balance);
        uint pull_amount = (99 * _pull_amount)/100;
        uint pull_fee = (1 * _pull_amount)/100;
        users[msg.sender].balance -= _pull_amount;
        users[owner].balance += pull_fee;
        payable(msg.sender).transfer(pull_amount);
```

```
      _totalSupply += pull_amount;
      emit pullKYCrypto(msg.sender, _pull_amount);

      return true;
     }


    /**
    @dev function the function Account_Blocking enables to block accounts
    @param _unlist the address to be blocked
    **/
    function Account_Blocking(address payable _unlist) public payable onlyOwner returns (bool success) {
      users[_unlist].status=Status.BLOCKED;
      emit blockedAcount(_unlist);
      return true;
     }


    /**
    @dev function the function Account_Unblocking enables to unblocked accounts
    @param _relist the address to be unblocked
    **/
    function Account_Unblocking(address payable _relist) public payable onlyOwner returns (bool success) {
      users[_relist].status=Status.APPROVED;
      emit unblockedAcount(_relist);
      return true;
     }


    /**
    @dev function SmartContractBalance get the balance of the smart contract back
    **/
    function SmartContractBalance() external view returns(uint){
      return address(this).balance;
    }


    //You can withdraw your cryptocurrency from the _withdrawKYCrypto function
    fallback () override external payable  {
      declinedusers[msg.sender] = msg.value - 10_000_000;
      emit unusedKYCrypto(msg.sender, msg.value);
    }


    receive() payable external override {}

 }
```

Appendix C: Fake Oracle smart contract

```
pragma solidity >=0.7.0 <0.9.0;
//SPDX-License-Identifier: UNLICENSED

interface Contract_interface {
        function Mailbox(uint32 _orderID, int88 _data, bool _statusFlag) external payable;
}
contract FakeOracle {

    uint orderId = 0;
    struct order {
        int64 commitmentID;
        string query;
        uint32 orderDate;
        uint40 _gasForMailbox;
        uint64 _gasPriceInGwei;
        address payable _addr;
    }

    mapping(uint => order) public orders;

    function GetTransactionCosts(int64 _commitmentID, uint40 _gasForMailbox, uint gasPriceInGwei) external view returns(uint transactionCost) {
        return 0;
    }
    function Order(int64 commitmentID, string calldata _query, uint32 _orderDate,uint40 _gasForMailbox, uint64 _gasPriceInGwei) external payable returns(uint32 orderID) {
        order memory o = order(commitmentID, _query, _orderDate, _gasForMailbox, _gasPriceInGwei, payable(msg.sender));
        orders[orderId] = o;
        orderId++;
        return uint32(orderId -1);
    }

    function relay(uint32 _orderId, int88 _data, bool _statusFlag) external {
        order memory o = orders[_orderId];
        Contract_interface c = Contract_interface(o._addr);
        c.Mailbox{value:0}(_orderId, _data, _statusFlag);

    }

    function returncurrentOrder() view public returns (uint256) {
        return uint(orderId-1);
    }


}
```

Appendix D: bcp_informed smart contract

```
//SPDX-License-Identifier: UNLICENSED
pragma solidity ^0.8.0;

interface BCP_interface {
    function GetTransactionCosts(int64 _commitmentID, uint40 _gasForMailbox, uint gasPriceInGwei) external view returns(uint transactionCost);
    function Order(int64 commitmentID, string calldata _query, uint32 _orderDate,uint40 _gasForMailbox, uint64 _gasPriceInGwei) external payable returns(uint32 orderID);
    function cancelOrder(uint32 _orderID) external payable;
}

abstract contract BCP_informed {
    BCP_interface BCP;
    address payable public BCP_Address;
    modifier onlyBCP {
        require(msg.sender==BCP_Address);
        _;
    }

    event ReceiverConnection(address Rec, address indexed SC);
    constructor(address payable addr) {
        emit ReceiverConnection(msg.sender,address(this));
        BCP_Address = addr;
        BCP = BCP_interface(addr);
    }

    function getBCPAddr() external view returns (address payable) {
        return BCP_Address;
    }

    function Mailbox(uint32 _orderID, int88 _data, bool _statusFlag) virtual external payable;
    fallback() virtual payable external;
    receive() virtual payable external {}
}
```