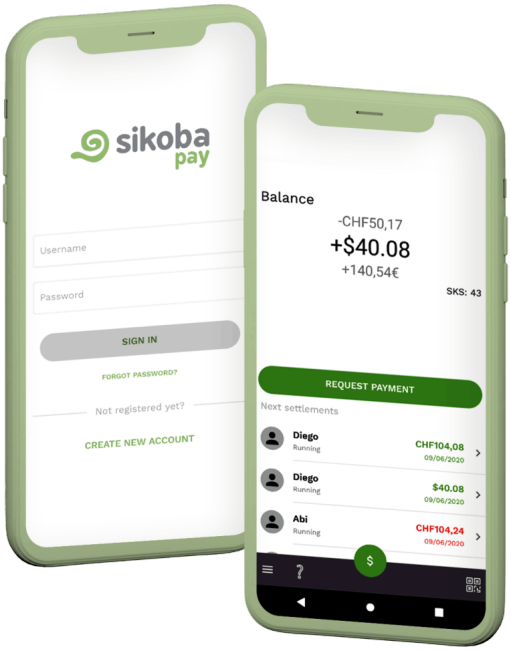
Sikoba Project Overview

Alex Kampa, Guillaume Drevon, Dragan Postolovski



The SikobaPay app is currently in beta testing,
we are targeting mainnet launch for the end of 2020.

v2.01 - 10 September 2020

Abstract

Sikoba Project Overview

Sikoba is a blockchain-based platform to register, track and clear debt. It has been developed to overcome the limitations of informal credit, which is widely used in developing countries to overcome money scarcity. IOUs (from "I owe you" - an acknowledgement of debt) are used instead of cash. However, informal credit does have certain limitations, including a lack of legal recognition, the inability to transact outside of one's trusted network an no verifiable audit trail.

Sikoba overcomes the limitations of informal credit:

1. Because debt is registered on a blockchain, users obtain legal recognition;
2. Users can transact beyond their immediate circle of trust, as Sikoba will automatically identify trusted intermediaries;
3. SikobaPay automatically clears circular debt, thus reducing the need for cash settlement;
4. SikobaPay produces a verifiable audit trail, and thus a credit history.

Sikoba can also be adapted to support many different applications, for example local currencies, mutual credit networks, micro-lending, basic income programs as well as digital cash programs.

This document provides an overview of the Sikoba credit system, its technical implementation as well as the project status.

Contents

List of Figures

Chapter 1

Introduction

Despite technological advances, a significant portion of the world's population remains unbanked[1]. According to the Global Findex database, there were *1.7 billion* unbanked people in 2017[1]. For these people, the way to financing is through non-regulated, non-supervised loans and credits.

1.1 Motivation

Throughout the ages, these circumstances have led people to adapt and innovate their own solutions to this crippling issue. For millennia, people have been using *informal credit* to reduce money dependency and boost their economies. However, all this was being done on "pen and paper", which in itself introduces other limitations for its user and is not as efficient as it could be.

Informal credit and its limitations are the main driver behind the development of the Sikoba platform. Our platform is designed to solve the inherent issues of informal credit when it's used in an "analog" manner.

1.2 The Sikoba project

The Sikoba project is a *decentralized finance* solution designed to facilitate the usage of informal credit, reduce money dependency and boosting local economies in regions in development.

[1] person that has no access to formal banking services

Planned as a web and mobile application with blockchain backend implementation, the platform aims at tackling the issues that informal credit has, while providing unquestionable security.

To do this, there are three key points to address: data security, instant intermediary identification and mutual debt clearing.

1.2.1 Secured data storage

By storing data on a blockchain, the system can provide authentic and indisputable proof of existence of IOUs to all parties involved, thus disabling any attempt at forgery and scam.

1.2.2 Intermediaries

Furthermore, by having data of users' inter-relations, the system can identify intermediary relations between users in real-time, thus saving users hours that they would otherwise spend searching for possible connections.

1.2.3 Debt clearing

Lastly, the platform will be able to solve the biggest issue, which is money dependency. By running cycle-detection algorithm, it is possible to identify circular relations between users, which can allow for debt to be cancelled, thus reducing overall money dependency.

Chapter 2

Money Theory

When developing a product for public use, the main priority is that it is helpful to its user base. To this end, before continuing, it is necessary to set a baseline and get familiar with important theoretical and practical concepts. These describe the financial and social theory on which informal credit usage is based and based on which, the system's requirements will be established.

2.1 Money primitives

Primarily, there are three ”primitive” monetary operations at the core of all financial interactions. These are:

1. **Issuance** - a money token(regardless of its form) is transferred from its issuer to another party - this is when ”money” is created
2. **Transfer** - a money token is transferred between two parties, none of which is the issuer
3. **Cancellation** - a money token is returned to its issuer, thereby ceasing to be money - this is when ”money” is destroyed/cancelled

2.2 Informal economics

Informal economy, also referred to as grey economy, is part of an economy that is not governed by the state or any authority for that matter, financial or otherwise[2]. As such, informal transactions

are not taxed and do not contribute to the official bottom line of national GNP[1] or GDP[2] numbers. A report by the International Labour Organisation estimates that more than sixty percent of the world's working population, roughly two billion people, works in informal economies[3].

From a financial point of view, informal economy is characterized with small scale operations, i.e. volume of a single transaction is relatively low. On the other hand, the number of transactions that occur is often elevated, which leads to an overall volume of transactions that cannot be ignored nor underestimated.

2.3 Informal credit

In the very core of informal economics lies *informal credit*. Product of unbanked societies that has been around for thousands of years, informal credit still holds its ground as the primary mean of financing and transactions for a large part of the world.

Mostly localized to third-world countries and countries in development, informal credit comes as a suitable alternative to conventional banking which is scarcely, if at all, available in these parts of the world.

2.3.1 Advantages

One of the main advantages of informal credit is the flexibility it offers when transacting. Most transactions happen on a trust basis and for two individuals to trade directly, it is necessary that they trust each other. As such, instead of settling the debt immediately with cash, the two parties can agree that the settlement be done at some point in the future,i.e. be delayed.

What this effectively does is it creates an IOU[3] from the buyer/debtor towards the seller/creditor, with a certain term and amount. As we see later, this is the primary property upon which certain improvements can be implemented on informal credit systems.

[1] Gross National Product
[2] Gross Domestic Product
[3] I owe you

2.3.2 Limitations

There are however conceptual difficulties related to the usage of informal credit. For one, since there is no formal proof of IOUs between two individuals, awkward situations can arise where one individual can pretend to ”ignore” his previously agreed upon IOU, or another user can insist on the existence of an IOU that has never been created.

Another problem arises due to the trust-nature of relations. Since individuals can only transact with other people whom they explicitly trust, issues arise when a transaction needs to be done between two persons that do not directly trust each other. The solution for this is to search for intermediaries between the two involved parties. This is usually done over the phone and is a process which can take *several hours* to successfully accomplish.

Last but not least, there's the issue of money dependency which, in the worst case, can cause deadlocks in informal credit interactions. If we imagine a cycle of three people, each of which has a debt of $100 towards the next user with the same term, in order to settle their debts, there will need to be a total liquidity of $300 between the three users. This issue can be crippling in some cases, due to the low amounts of cash circulating in third-world countries. If we ”clear” this debt, however, the users would need no cash whatsoever and their debts will be completely settled.

2.4 Credit Theory of Money

The Credit Theory of Money is an overarching term which encompasses multiple theories whose common idea is that money is essentially a claim on its issuer, and therefore a specific type of debt. While this is obviously true in modern banking systems, proponents of the Credit Theory maintain that the credit aspect of money has almost always been present, including during Antiquity and during times when there was a metallic standard. Important proponents of the Credit Theory were Henry Dunning Macleod and Alfred Mitchell-Innes [4].

This Credit Theory is also supported by David Graeber, an anthropologist who wrote the book *Debt: The First 5000 Years*. In his book, Graeber claims that based on the evidence he found while researching, it is more likely than not that ancient monetary systems were debt-based.

2.4.1 Money creation

An important notion which connects the credit theory of money and physical money as they are commonly used in the world is the process of money creation.

A long-running belief is that all money in circulation is being created by central banks, from which they are then slowly trickled down to private banks and institutions. In this context, the term *fractional banking* is used, since it is presumed that the amount of money being distributed to banks is a fraction of the total money that the central bank produced. This, however, is wrong.

Money does not get created *exclusively* at central banks. While they themselves create money, the amount they create is but a fragment of the total volume of money that is created. Most of today's money is being created by private banks.

When a loan is granted to a customer, what is given to him by the bank is technically "new" money. This will not change the customer's overall wealth, since he will also have a debt for the same amount. The situation is similar for the bank. To counter the negative outflow, the bank needs to possess some asset whose worth would balance out the negative outflow caused by the loan. With this, there's now new money put into circulation. It did not exist before being credited to the customer. Of course, this has the consequence that as the loan is being paid off by the customer, the money loaned to him is being removed from the bank's balance sheet, thus it's being "destroyed", or "cancelled".

2.4.2 Credit conversion

The way money is created in the system can also be used to describe another phenomenon of credit transactions.

In essence, what happens is credit exchange, or the more context-accurate term, *conversion*. The bank, whose credit is almost universally accepted, agrees to accept the customer's lesser credit and in turn exchange it by giving him its own. Since now the customer has the bank's credit, he can use it with anyone who accepts it. And since the bank's credit, as mentioned, is almost universally accepted, it can be used for anything and anywhere.

If the customer were to try giving *his* credit instead, to say, other businesses, he would most likely get laughed at, since his credit is not as universally accepted as the bank's.

2.4.3 Debt clearing mechanics

Using cashless transactions, i.e. credit promises for delayed payments instead of on-the-spot settlement, creates several seemingly unnatural phenomena. One such phenomenon is what is called *time mismatch*[5]. A trader can be found in a situation where he has a net positive balance, yet he does not have enough liquid assets to repay his outstanding debts. If one trader gets into this state, it can potentially have a domino effect on his wider circle of partners, which can eventually cause a ”gridlock” - it can prevent any further trade due to liquidity constraints.

Going back to the Middle Ages in Europe, a mechanism was created to handle this potential issue. Known as *rescontre*, *skontrieren* or *vivre compte*[5][6], this mechanism allowed merchants to clear their liabilities. During the procedure, the merchants would get together and they would name their debtors, and the named creditors would confirm the stated liabilities. In the first stage of the process, a merchant would try to offset reciprocal credits and debts with his debtors. In the next step, attempts are made to identify cycles between merchants on which debts can also be cancelled. Once no more cycles can be found, all remaining debt had to be settled in cash[5].

For the merchants, this helped prevent trade deadlocks. Today, this mechanism can be used in a similar way to reduce money dependency for users of informal credit.

Chapter 3

Sikoba Credit System

The entire edifice of the Sikoba credit system rests on the concept of credit acceptance. The desire of an economic agent to issue credit means nothing without the presence of other economic agents willing to accept that credit.

3.1 Credit acceptance overview

In the normal, unconstrained course of business, credit acceptance is voluntary. Governments can however impose laws of legal tender. Legal tender money is generally backed by government credit which is the highest credit available and would be accepted even without legal tender laws[1].

3.1.1 Specific credit acceptance

In general, before a credit relationship can exist, economic agents need to know each other and have an established trust relationship. The decision to accept a counter-party's credit is then based on specific information about that counter-party's circumstances and creditworthiness.

[1] it is useful to keep in mind that a) legal tender laws are of no material importance to a functioning money system, viz England 1979-1820 and China for most of its history b) it was not unusual for kings and princes to devise draconian punishments to force an unwilling

3.1.2 High-credit institutions

Certain organisations acquire a notoriety such that their credit becomes widely accepted in the marketplace. Such organisations have generally obtained a credit rating from one or more reputable rating agencies, which enables them, for example, to issue publicly traded debt instruments. Many economic agents will be prepared to accept the credit of such high-credit institutions based on their credit rating rather than on a direct trust relationship.

3.1.3 Money-grade institutions

Banks are a special case of high-credit institutions because, by definition[2], their credit is money and is therefore universally accepted. In most countries, the credit of the central government and of certain government agencies is of equal, if not higher, creditworthiness.

3.2 Credit acceptance in Sikoba

The basic concept in Sikoba is simply this: a user can choose to accept another user's credit, or IOU(s). That credit line will need to have certain mandatory characteristics, such as currency and amount. Let's say Bob decides to accept Alice's credit up to the amount of €500. This means that he is willing to accept Alice's IOU as payment, instead of money, up to the amount of the credit limit. This representation can be seen in figure 3.1.

FIGURE 3.1: Credit acceptance - baseline.

To be more precise, we can include the credit limit and the indication that there is no outstanding obligation yet, as shown in figure 3.2.

In this relationship, Bob is the **"credit acceptor"** (potential creditor) and Alice the **"credit issuer"** (potential debtor). It is important to note the fundamental asymmetry in the roles of Alice and Bob. Bob is active at the beginning: he decides to accept Alice's credit. Then Bob and Alice negotiate credit terms. Once that is done, Alice has the initiative: she can decide when to use the credit line.

[2] "by definition" because when a bank's credit declines so far that it is no longer money, then it ceases to be a bank.

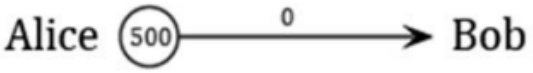


FIGURE 3.2: Credit acceptance - no obligation.

Note that this asymmetry will even out in the case of bilateral credit agreements. Suppose Alice visits Bob's shop and decides to buy a widget costing €200. Of course, Alice could just pay Bob in cash if she wanted to. But she could also choose to use her credit line, in which case the result can be represented as shown in figure 3.3.

FIGURE 3.3: Credit acceptance - with obligation.

3.2.1 Negotiation of credit terms

A credit line between two users is created following a negotiation process. Suppose Bob and Alice do not yet have a credit relationship in Sikoba, and Bob wants to grant Alice a credit line. Before definitely creating the credit line, Bob and Alice will need to mutually agree on the credit amount, repayment target, interest and various other metrics which will dictate the IOU creation process.

3.2.2 Modifying credit terms

Modification of credit line terms by mutual consent will of course be possible. From a technical perspective this will, with some minor exceptions, mean closing an existing credit line, reopening another, and similarly closing/reopening any relevant IOUs. When the terms of the new credit line are, in all of the possible aspects, equal or more favourable to the credit issuer, the procedure may be done unilaterally by the credit acceptor.

3.2.3 Cancelling credit lines

A credit line can be cancelled by any of the two parties, as long as there are no IOUs still running on the credit line. If that is the case, the IOUs will first have to be settled, before cancelation is possible.

3.3 Credit conversion

In the Sikoba system, as in the real world, users will only be asked to accept the types of credit that they specifically choose. Money, being the highest credit, is accepted by all, and there may also be some other well-known entities whose credit will be widely accepted. But otherwise, if a user could only have credit interactions with his trusted network, this would significantly limit the number possible of credit interactions in the system. For that reason, we use the age-old method of going through trusted intermediaries. In Sikoba, we call this mechanism *credit conversion*. Credit conversion occurs when one participant allows another to use one of his credit lines. If Alice wants to pay Charles, but does not have a direct credit relationship with him, she may be able to use the credit line that Charles has granted Bob.

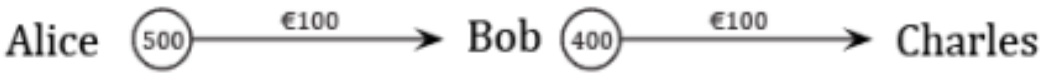


FIGURE 3.4: Credit conversion.

Applying the mechanism of credit conversion successively, Alice gains the ability to transact with many users even if she does not know them. Users have full control of the credit conversion process. For each credit line granted to him, a participant will be able to decide whether that credit line will be available for credit conversion, to whom, up to what amount and on what terms. In the example above, Bob may have decided to make available to his network €200 of his credit line to Charles, for at most 30 days and at 1annualised interest. Note that Bob would have made this decision beforehand. He would not need to take any action when the transaction between Alice and Charles occurs.

3.4 Clearing

Clearing consists of finding closed transaction loops that reduce the overall cost of credit in the system. When two states with the same cost can be achieved, then the state with the least outstanding credit will be preferred. In the following simple example, exposures totalling €10,500 are reduced to a mere €1,000 via a single clearing cycle.

For a clearing cycle to be valid, the overall combined cost of the credit exposures should be lower to or equal to the cost before clearing (the issue of defining cost is a separate issue.) We can have

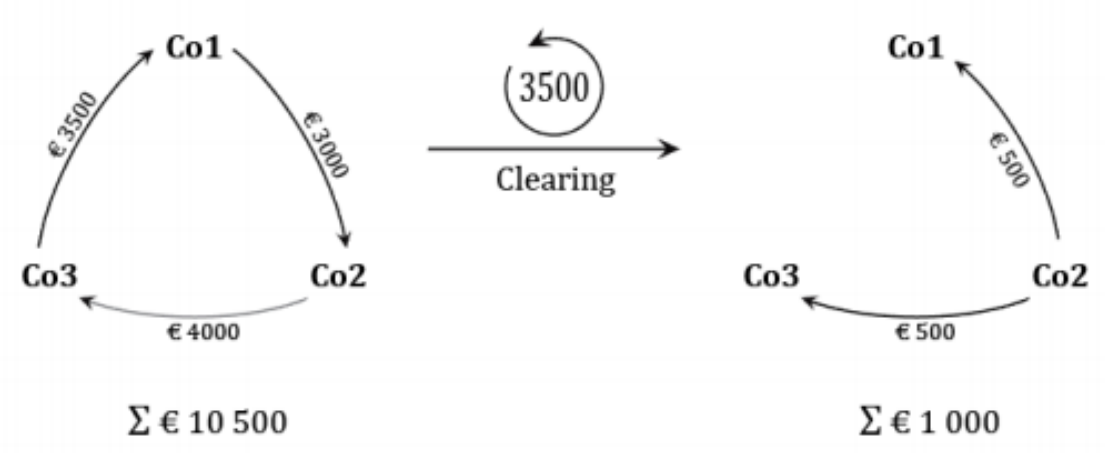


FIGURE 3.5: Clearing debt 1.

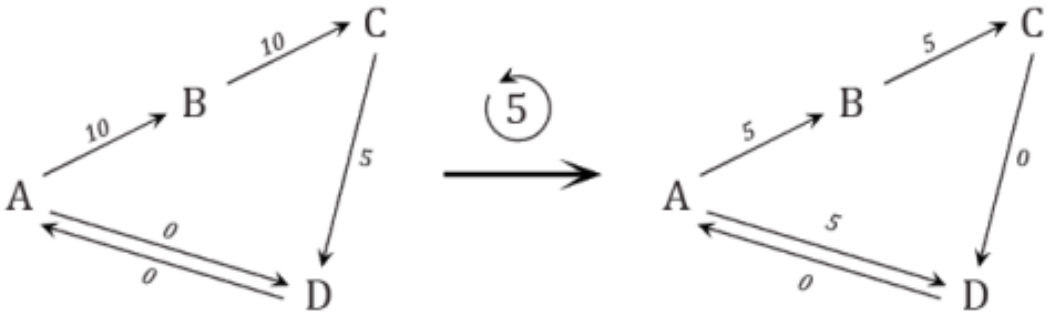


FIGURE 3.6: Clearing debt 2.

situations where the overall cost goes down, but one or more users in the clearing loop start paying more. In the example in figure 3.6, if A pays more for the exposure to D than to B, then the result of the clearing may be that B and C pay less but A pays more. There are two possible ways to deal with this:

1. We do not allow such clearing cycles, or
2. We introduce a mechanism by which the ”losers” of the clearing are compensated by the “winners”, so that everyone ends up a winner.

Because of the potential complexity involved, the first option may be chosen at first, but with the possibility of implementing a compensation mechanism later. Also note that there will be

optimisation issues, as one can easily imagine situations where starting with the clearing at the wrong place will make further clearings impossible. For example, doing a small clearing loop first may prevent the system from doing a much larger loop later. Multi-currency clearing will also present a challenge.

Chapter 4

System design

Having gone through the analysis process, this chapter explains the design process for the system, taking into consideration all the technical and functional requirements that the platform will require.

4.1 System architecture

From a top-level point of view, the system consists of a simple network of nodes. The nodes communicate between themselves, which allows for a continuous data exchange between them, which in turn enables the blockchain layer of the platform.

4.1.1 Nodes

Nodes are at the core of the sikobaPay infrastructure. Ran individually by highly trusted parties, nodes take part of the blockchain consensus, which is what coordinates the transactions that every node executes in this decentralized setting. In addition to active nodes, there will also be gateway nodes which will do load balancing, as well as passive nodes which will be used for simple data-retrieving queries, all with the goal of optimizing performance at the main nodes. The 'Scalability' section goes into more detail on this topic.

4.1.1.1 Main nodes

Each main node runs a copy of the main application together with the latest data of the system, as well as a blockchain middleware component. The nodes use the middleware to coordinate the transactions that they generate between themselves and then apply each transaction to the database.

The structure of a main node is shown in figure 4.1.

4.1.1.2 Passive nodes

Passive nodes serve as read-only nodes. Their purpose is to answer to queries sent by users. In order to do this effectively, passive nodes need to periodically update their database to keep up to date with the main node network. By assigning this task to these nodes, a significant amount of stress is taken off the main nodes, thus enabling a more performing system.

4.1.1.3 Gateway nodes

Gateway nodes are the coordinators within the system. They receive client requests and route them to one or more main nodes for processing. By analyzing stress and load at main nodes, they decide which main node gets to process which transaction request, with the goal of reducing latency and increasing performance. Additionally, by analyzing passive node behaviour, gateway nodes decide which passive node gets to serve which query request. This is yet another step to increase performance within the system. Finally, gateway nodes also protect the network from DDOS attacks.

4.1.1.4 Clearing nodes

Clearing nodes maintain their state synchronised with main nodes, and continually work to optimise and re-balance the system's IOUs by finding clearing cycles and submitting clearing proposals to the main nodes.

4.1.2 Network

The Sikoba Network is build as a classic state machine replication system. The Sikoba core system, which acts as both the state machine transition and output function, is deterministic and handles

transactions sequentially. A blockchain layer can therefore easily be added to the system, to enable decentralisation and Byzantine fault tolerance.

There are several communication channels on the network. First, there is the inter-node communication, which is necessary to enable metadata sync between nodes. Then, there is the blockchain middleware communication that happens on the same network. The middleware component at each main node will communicate with other middleware components to enable the blockchain consensus. Additionally, there's the communication between the application layer and the blockchain middleware. This channel transmits the transaction submission requests from the application layer and the block requests from the middleware. The network is partially open to the outside world, as only the gateway nodes will directly communicate with the platform users.

Figure 4.2 shows a simple top-level description of the sikobaPay network.

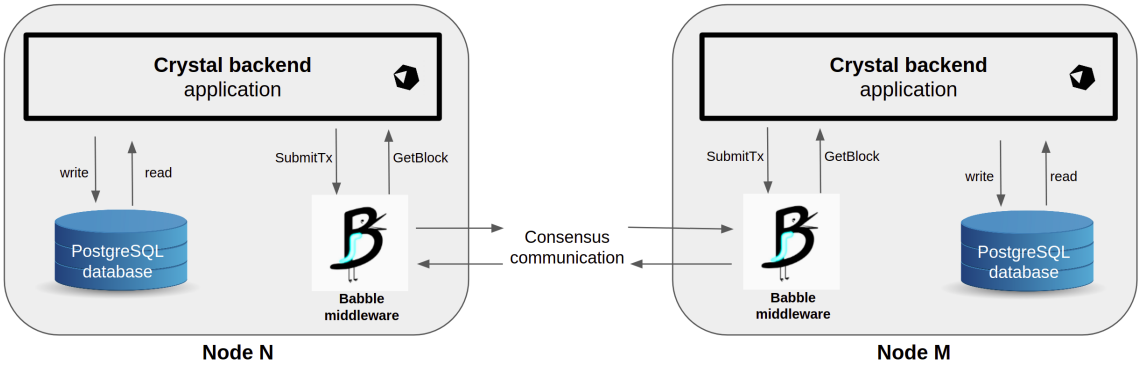


FIGURE 4.1: Nodes structure and communication.

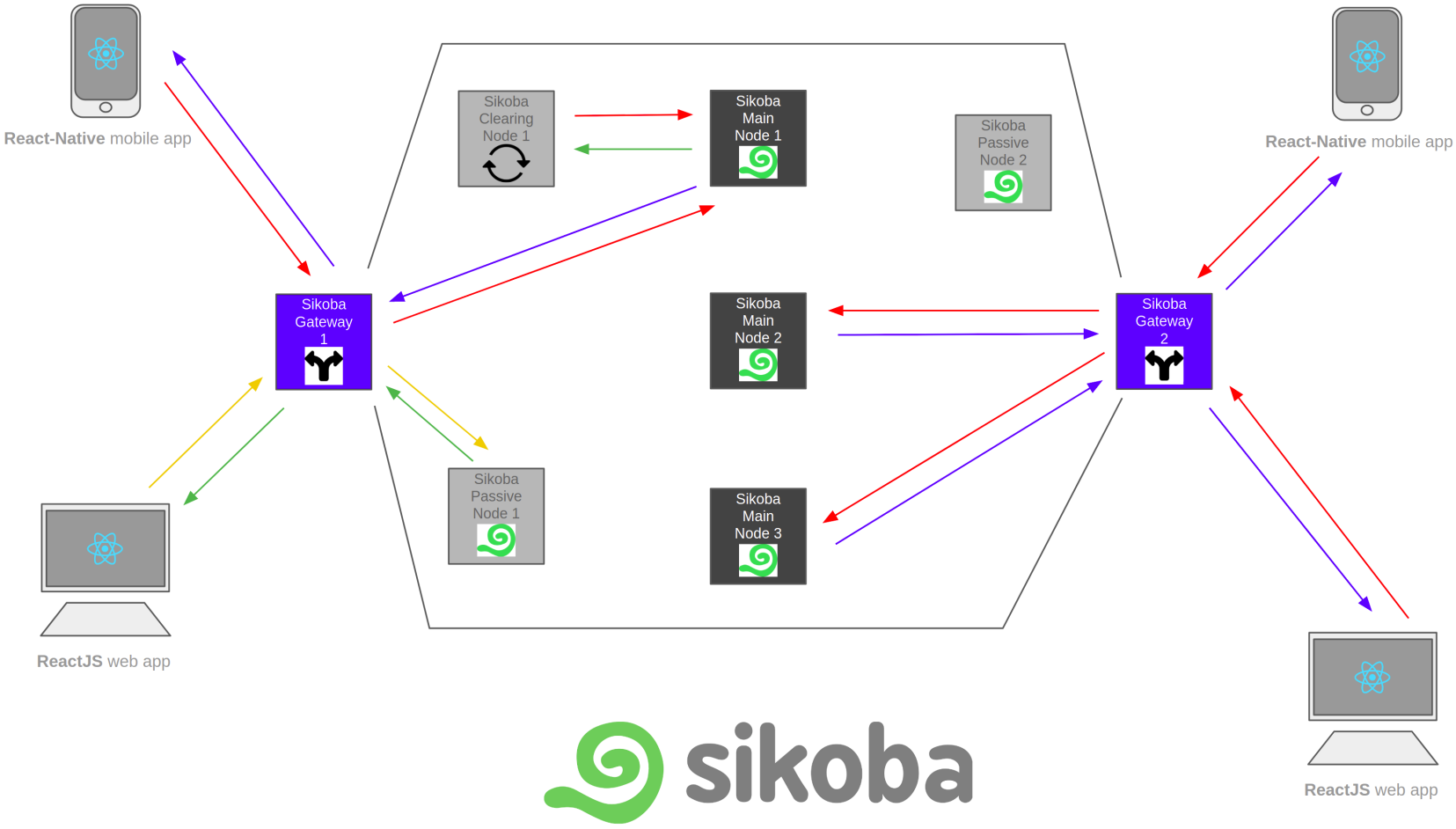


FIGURE 4.2: System architecture.

4.2 Application layer

The application layer contains the business logic that runs the system and defines possible behaviour and functionality.

Users will be able to use Sikoba both on mobile devices as via web browsers. Usage on mobile devices will grant ordinary users easy access to all the functionalities they will require from the system, while the web application will allow companies and businesses more flexibility when processing bulk data.

4.2.1 Important data models

There are some data models which are necessary for the platform to work.

4.2.1.1 User

A *User* object represents data for one unique user of the platform. It stores personal data of the user, as well as preferences of the user related to the system. Since there's no functionality available for anonymous agents, every user that wants to use the system will have to register.

4.2.1.2 Currency

A *Currency* object represents a real-world currency. Each currency should have attributes that contain data mostly related to the way amounts in this currency are represented to the end-user. Initially, currencies can only be added by the system, although at a later stage, it is planned to enable user groups to create their own local currency and add it to the system/their circle of interaction.

4.2.1.3 Credit line

A *Credit Line* object represents a directed relation of *trust* between two different users in the system. Once a credit line has been offered and accepted, the credit issuer can start creating IOU payments towards the credit acceptor with amount, term and other limitations being mandated by the attributes of the credit line. Credit lines can also be permanently cancelled, either by the credit issuer or the credit acceptor.

4.2.1.4 Conversion permit

Credit lines by themselves only directly connect two users. This is quite limiting because for any two users to be able to pay one another, they must explicitly *trust* each other, which means that every user in the system will have to be trusted by any other user whom he would like to pay. This is not a realistic expectation and as such, this model would not be very effective nor helpful.

That is the issue that conversion permits help solve. A *Conversion Permit* object represents a secondary directional trust relation between two users, for a given credit line to a third user. With permits, users can allow other users to use credit lines that have been directly granted to *them*[1], even though there's no relation between the permit grantee and the credit lines in question. This enables *credit conversion*.

4.2.1.5 IOU

An *IOU* object represents an owing relation between two users. IOUs are the core of the platform, since they are also at the core of informal credit transactions. An IOU object has a *base amount*, an additional *fees* amount, a term, or as referenced in the system, a *target*, and other minor attributes. IOUs can be either created explicitly by users, or by the system as a result of a clearing process. Additionally, IOUs have an attribute named *refundable fee* and a calculated attribute named *current refundable fee*. To explain them, it needs to be stated that IOUs, despite having a due date, can be settled earlier. Since the potential clearing date is not known at the time of creation, the payer will need to cover all fees upfront, including interest fees that will be calculated for the whole potential term of the IOU. The default *refundable fee* has this value, while the calculated *current refundable fee* will show the amount that would be refunded if the IOU were cleared on a given date.

In the system, an IOU can only be marked as settled by the credit acceptor of the credit line on which it was created.

4.3 Blockchain

Bitcoin was the first completely decentralized and widely successful application. It started in 2008 and is still used extensively. It was the first public blockchain at the time, allowing the trading of a digital currency without any central authority. In 2014, Ethereum added smart-contract features

[1] them being the users that grant the permits

to its blockchain and created the first decentralized computer. Many other blockchains have been created since, but none, including Ethereum and its Turing-complete language, allow in practice to run complex decentralized applications.

However, some blockchain technologies that have recently emerged allow applications to implement their own custom blockchain by providing a *blockchain middleware* that can be plugged into the application. This is what Sikoba aims for; to use a trusted and robust blockchain layer while keeping full control over the business logic of the application.

4.3.1 Blockchain as middleware

Blockchain middlewares have an API to interact with, allowing the application to send transactions, and in return, retrieve blocks of transactions. Since the middleware is responsible of running the consensus mechanism on its own, the application layer does not need to know which consensus mechanism is used nor how it works.

The Sikoba application stores the system data in a database locally at each node. All reading requests(queries) can be run in a normal manner. However, once the user wants to change some data in the system, whether that's changing his personal information or creating a payment, this request has to be wrapped in a transaction object. These transactions are the core of the blockchain layer. They are sent to the middleware via the API which then runs the consensus and builds blocks. Once a block is received, the application layer splits down to individual transactions and executes and applies every one of them.

4.4 Anchoring Sikoba transactions

Because debt/credit registered on the Sikoba blockchain is financial data, our system needs to be fully reliable in terms of data management. This is the reason why we are building a blockchain storing transactions between users. Blockchains are well known for their redundancy and immutability properties.

In addition, the nodes run by the several institutions, members of the Sikoba Federation, provide decentralization and prevent a single entity from doing any malicious activity.

Finaly, transactions between users are highly private and must remain within trusted nodes only, so we also require to use a private blockchain. But in that case, how can we guarantee the correct behavior of the system if everything remains private?

The answer to that question is: anchoring.

4.4.1 Basic Anchoring

In order for an individual to trust that debts and credits are properly registered, we want to integrate our system with a public blockchain. This will be done by registering a Merkle root of every block of Sikoba's blockchain on a public blockchain. That way, we can generate the corresponding Merkle path every time a transaction is inserted into a block. A user can verify his transaction has been correctly registered by Sikoba using only the Merkle root on the public blockchain and the Merkle path provided by Sikoba.

Only Sikoba should be able to update these Merkle roots on the public blockchain. This can be done in several ways, for instance using a smart contract controlled by Sikoba, or by having the public blockchain verify that the data comes from Sikoba using some signature scheme. Precise specifications will depend on the public blockchain's capabilities.

4.4.2 Strengthening the Anchoring Process

In order to strengthen the anchoring process, and to remove any possibility of Sikoba registering incorrect information on the public blockchain, we could do the following:

- For block B, register the Merkle root M of transactions contained in B, along with hash(B).
- When we want to register the next block B':
  - we provide M' and h' = hash(B')
  - we also provide a ZKP proving B' is linked to B, by proving that we know some transactions T' such that Merkle root of T' = M', and hash(h,T') = h'

That way, we ensure that the data we provide comes from a valid chain, without disclosing any individual transactions. The transactions will be treated as private input variables of the ZKP.

In case several blocks need to be synchronised simultaneously with the public blockchain, we could use proof recursion.

4.4.3 Proofs of Transaction History

In addition to proving that transactions were correctly registered on the Sikoba blockchain, a user may also want to use the public blockchain to prove the value and term of a credit balance he has with another user, or to prove his credit history.

For instance to prove a credit balance with another user, a user can aggregate the transactions he has with the other user and generate a ZKP that he has a Merkle path of the transactions corresponding to some Merkle roots registered on the blockchain by Sikoba, corresponding to a term/value. He can thus provide a publicly verifiable proof of a credit balance without disclosing any transaction details. Generating such proof could be a premium feature that would generate transaction fees both for the Sikoba Network and for the partner blockchain.

Note that a user, by carefully selecting transactions (for instance not taking into account repayments), could generate a proof for a wrong (or to be more precise: incomplete) value. This is not a significant problem as this attempt at cheating can be made visible later on, while remaining stored forever on a public blockchain. The other user will be able to generate a proof with the correct value, contradicting the original proof. The fact that this proof is more complete can be seen by looking at the number of transactions included. It is possible to keep this number private if needed.

4.4.4 Expected Volume of Anchoring Transactions

Block times on the Sikoba blockchain are expected to be very short, so anchoring will not be on a real-time basis. We will aim to write basic anchoring transactions once every 5 minutes. More complex anchoring transactions that include ZKP can be added later and do not need to be done as often. As to user-generated proofs, these will be initiated by and mostly paid by users, their volume should increase as the number of Sikoba users grows.

4.5 Scalability

Sikoba aims to eventually reach tens of millions of active users. To be able to support such numbers, the network therefore must be massively scalable. There are several critical sections in the system where optimization can yield significant performance boosts to help reach this goal.

4.5.1 Current scalability

Currently, performance optimization is achieved by separation of concerns of the business logic. By having different types of nodes in the network, we can limit the tasks and functionality that each node type needs to support. In other words, we have specialist nodes; clearing nodes run the clearing algorithm, passive nodes return data, main nodes execute transactions, etc.

Additionally, the gateway servers provide further performance boost by acting as request filters, so that the main nodes are not overloaded with bad requests.

4.5.2 Scaling Sikoba in the future

The previously mentioned solutions are only a drop in the ocean. There's plenty more to be done to ensure a scalable platform. Other possibilities for future scaling are:

- Improve consensus
- Sharding
- Main nodes infrastructure

4.5.2.1 Improve blockchain consensus

If necessary, the BFT parameters can be relaxed to achieve "one-step consensus" in most cases. Since the nodes are ran by trusted parties, there is no particular need to tolerate up to 1/3rd Byzantine nodes. This can help reach consensus faster, thus speeding up the entire system.

4.5.2.2 Sharding - sub-nets and side channels

To further speed up processing of requests, main nodes can be configured to run on a subset of the data. More precisely, nodes can be made to only work with transactions in a given currency, in a given location or with a subset of users and user groups. Dedicating nodes like this will provide more efficient data processing and transaction execution.

4.5.2.3 Main nodes infrastructure

On the hardware side, the simplest solution would be to max out all nodes with storage and processing power, and also place the main nodes in top-tier data centers, thus ensuring top-notch performance and low-latency network communication.

Chapter 5

Cryptography

5.1 Primitives

5.1.1 Cryptographic Hash

Cryptographic hash functions are deterministic algorithms that map data of arbitrary size into a fixed size value. Given the resulting hash value, it is in practice impossible to find a pre-image, except with brute-force methods.

SHA256 has been designed by NSA in 2001 and produces 256-bits values. It is extensively used in Bitcoin blockchain.

Argon2 is the winner of the Password Hashing Competition in 2015. The algorithm enforce some time/space complexity required to run it so that brute force attacks are made impractical.

Merkle Tree are cryptographic accumulators based on a hashing function. It is usually implemented as a binary tree where a node contains the hash of this two children. This data structure is used a lot in blockchains, where the Merkle root of the block's transactions is added to the block header.

5.1.2 Encryption

One-Time-Pad is the only encryption that cannot be cracked. It is a symmetric-key algorithm where the key has the same size of the message but cannot be re-used for another message.

Aes128: AES is an encryption standard specified by NIST in 2001, it is a symmetric-key encryption algorithm.

5.1.3 Digital signatures

Digital Signatures are public-key cryptographic schemes that allows a user to ensure a message has been generated by him, the owner of a private key. Recipient can verify the signature using the user's public key.

ECDSA is a digital signature scheme based on elliptic curves, it is a standard approved by NIST and used in Bitcoin and Ethereum.

Secp256k1 and **Secp256r1** define elliptic curve parameters for use in cryptography schemes such as ECDSA.

Diffie-Hellman key exchange is a way to exchange securely a shared secret. It is based on the discrete logarithm problem and allows two entities, by sharing only their public keys, to compute a shared value using their own private key.

5.1.4 Zero Knowledge Proofs

Zero knowledge proofs (ZKP) are cryptographic schemes allowing a prover to provide a small and efficient (probabilistic) proof he executed a publicly known computation, using some private inputs.

They are usually used in blockchain context to provide transaction confidentiality or to improve performance by outsourcing some computations off-chain in a trusted way. isekai is an open-source project from Sikoba that allows regular programmers to use zero-knowledge proofs in their application, using popular ZKP libraries.

5.2 Key Management

5.2.1 Master Key

A user is represented by a public master key, for which the user holds secretly the private key. The secret master key is only used at registration to generate the account number and what we called

device keys on client side. The master key is generated on client-side from a seed phrase and an optional password, following Bitcoin BIP39 [7] specification . It uses ECDSA scheme and elliptic curve secp256k1. Users can keep full control of their master key, but non-expert users can use Sikoba to store it securely. User master keys are safely encrypted and cannot be retrieved without user input.

5.2.2 Account Number

The account number is derived from the master key using Argon2 [8] hash of the master public key. It is 20 characters long, starting with SKO and ending with Fletcher's checksum. Here is an example:

SKOA-K0FP-RUNT-X3W0-FU0M

The size of the account number guarantees 75 bits of security, and security is enhanced with the use of Argon2 hash. Furthermore, the sign-up transaction being recorded in the blockchain, the account number can never be linked with another key. The device keys are used to authenticate a device and to sign regular Sikoba transactions.

5.2.3 Device keys

Contrary to traditional blockchains (e.g bitcoin), the device keys are not derived from the master key using Hierachical Determinists wallets [9]. The reason is that this scheme is designed for bitcoin where one needs to generate many keys (typically one per transaction) so that his transaction history cannot be (easily) found. Depending on the case one must use hardened keys to ensure proper security and the leakage of a private key is problematic as it threatens all the children keys.

Sikoba on the contrary does not require a unique keypair per transaction as the blockchain remains private. We take advantage of this by generating random ECDSA secp256r1 keys as device keys. They are linked to the master key because the master (or another registered device key) signed the device key registration transaction. That way, leaking a private device key has not impact on the security of the other keys. Users can revoke the keys and generate a new device key when needed. The device key is stored on the device using built-in device safe storage (keystore/keychain for Android/iOS), or using One-Time-Pad in the web application.

5.3 Key recovery

5.3.1 Manual key recovery

Master Key recovery can be done by the user on the client-side thanks to the seed phrase used to generate the master key. If the user used the additional password at that time, he can safely backup the seed phrase and he just has to remember the password. If he did not use a password, he should keep the backup secretly.

5.3.2 Automatic key recovery

If the user did not want to take care of the key recovery himself, he has the option to store his master private key securely on our servers. The key is encrypted using AES128 encryption, using a secret key that is never stored on the server. This means that even if somebody hack a Sikoba node, he will not be able to retrieve master keys. The master key is encrypted on client side with a secret key derived from a Sikoba public key and a random user key, using Diffie-Helman key exchange. That way all master keys are encrypted with a different secret. The Sikoba private key used to recover the master key is not stored on Sikoba Server.

5.3.3 Community authentication

When a user needs to recover his master key from a Sikoba node, he would first need to authenticate. This is done with a kind of knowledge-based authentication, taking advantage of user's community. The user will have to answer challenges regarding information from his close contacts. For instance he will be presented to some photo and he will have to say if he knows this person or not.

5.3.4 Error-tolerant knowledge-based recovery

Once the user is authenticated, he will be asked other questions that only him knows the answer. In particular Sikoba nodes will not know the answer, contrary to the questions asked during the community anthentication. The questions are first asked at registration time and the answers are used to generate yet another password to secure the master key. One can see that the master key is protected jointly by a secret that only Sikoba support can use and a secret only known by the user.

Finally, the user does not need to remember all the questions because his answers are decoded by a Reed-Solomon error-correction decoding that allows for one-third of incorrect answers.

5.3.5 Community Recovery

We also plan to add a community recovery mechanism, in order to spread the user seed phrase to Sikoba and its community. The seed phrase would then be well secured because it cannot be retrieved without the joint effort of user's community and Sikoba. However, the user would have to trust that its community backup properly their part.

Chapter 6

Tokens

6.1 The two types of Sikoba tokens

Sikoba tokens ("SKO") represent an electronic form of value which will be used to (indirectly) pay for transaction fees on the Sikoba network.

SKO is the main Sikoba network token. Its price will fluctuate based on supply and demand, and tend to follow the transaction volume on the Sikoba Network.

SKS is the internal Sikoba Gas token, an accounting unit whose value will be fixed against a basket of currencies, initially against XDR [1] with 1 XDR = 150 SKS. All transaction fees on the Sikoba network will be expressed in SKS, making them stable and predictable over the long term.

SKS can be purchased using SKO and will be transferable between accounts, but cannot be exchanged back to SKO.

6.2 Current and future form of tokens

Because the Sikoba main-net has not yet been launched, tokens can currently exist only as temporary representations of future tokens, or as promises of such tokens. This can be compared to trading on a when-issued basis.

[1] Special Drawing Rights

Until late June 2020, a total of 1.51 million Sikoba presale tokens (symbol: SKO1) existed on the Ethereum blockchain at the following address:

https://etherscan.io/token/0x4994e81897a920c0fea235eb8cedeed3c6fff697

At the end of June 2020, a new ERC-20 contract with Sikoba tokens (symbol: SKO) was deployed:

https://etherscan.io/token/0x6B40089e6CBa08696D9ae48F38e2b06fAFF81765

The SKO1 tokens can be converted into SKO tokens, and most of them have already been converted. After Sikoba main-net launch, all token holders will be able to exchange these Ethereum SKO tokens for the definitive SKO tokens which will exist natively on the Sikoba blockchain.

6.3 Token economy

The demand for SKO tokens will be driven by users who wish to transact on the Sikoba network. Such users will have the possibility to purchase SKO tokens for fiat money or cryptocurrencies using various methods: on crypto exchanges, online using their credit card, from other Sikoba users, from local re-sellers such as participating stores, as well as from Sikoba local partners. We will also provide a way for Sikoba users to buy SKS directly, via a purchase of SKO followed by an immediate conversion to SKS.

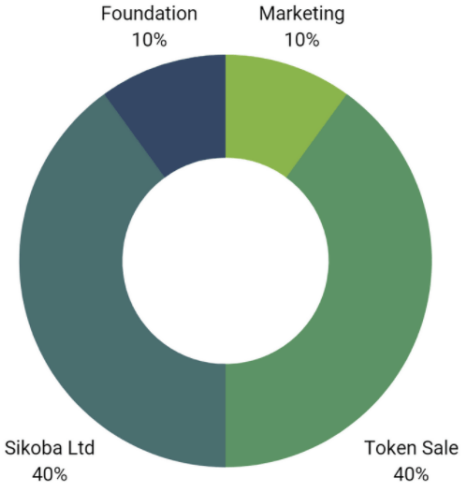
The Sikoba network will include an Internal Special Purpose Exchange ("ISPE") acting as an automated market-maker, fee collector and fee redistributor. The ISPE will tend to accumulate a net position in SKO, which it will periodically distribute to:

- The Sikoba Federation members (i.e. the node operators), to compensate them for the running costs of the nodes that they operate;
- The Sikoba Foundation, to provide it with resources with which to fund projects that will benefit the ecosystem;
- Sikoba network users, as incentives, e.g. as a reward for activity or staking.

The Sikoba Federation members and the Sikoba Foundation will tend to sell SKO tokens for fiat to fund their operating costs, thus closing the circle. Given a fixed supply of SKO tokens, this will naturally result in a tendency for the price of SKO to follow the volume of transactions on the Sikoba network.

6.4 Token distribution

The maximum token supply has been set to 100,000,000 and the planned token distribution is as follows:



- 40% retained by Sikoba Ltd;
- 10% reserved for the (not yet created) Sikoba Foundation;
- 10% for marketing;
- 40% of the tokens to be sold;

6.4.1 Sikoba Ltd tokens

The 40m tokens earmarked for Sikoba Ltd will be locked, with the 5 release dates being months 24, 30, 36, 42 and 48 after planned mainnet launch. Thus, 8m tokens will be released in each of: January 2023, July 2023, January 2024, July 2024 and January 2025. Independently of these release dates, Sikoba Ltd expects to remain a long-term holder of these tokens.

6.4.2 Sikoba Foundation tokens

The 10m tokens earmarked for the Foundation will be locked, with the 5 release dates being months 12, 18, 24, 30 and 36 after mainnet launch. Thus, 2m tokens will be released in each of: January 2022, July 2022, January 2023, July 2023 and January 2024. Of the tokens held by the Foundation, 70% will be put into a strategic reserve fund, to be used only in exceptional circumstances. The remaining 30% can be used to support the Sikoba ecosystem.

6.4.3 Marketing tokens

Approximately 425k tokens have already been spent on marketing or issued in lieu of payment, and a further 275k are earmarked, including for an airdrop. The remaining 4.3m tokens will gradually be used over the next 5 years for:

- 1m local marketing partners (most with 6-12 month locking period);
- 2.3m bonuses for Sikoba team members and advisors (most with 12-24 month locking periods);
- 1m for market-making.

6.4.4 Tokens for sale

Tokens earmarked as ”for sale” are distributed as follows:

- 0.7m already sold;
- 2m (maximum) for August/September 2020 private sale, with 3 months lock;
- 10m for IEO;
- 5m for IEO reserve, to be used in case of strong demand, otherwise to be locked;
- The remaining approx 22m tokens to be locked for at least 12 months, for a future round;

6.4.5 Available supply

Based on the preceding, we can estimate the freely circulating token supply after listing/IEO will be around 14.4m, consisting of:

- 1.4m previously sold and issued for marketing purposes;
- 2m from 2020 private sale;
- 10m from IEO;
- 1m for market-making;

Chapter 7

Sikoba Foundation and Federation

At the launch of the Sikoba mainnet (i.e. the production version of the Sikoba network), Sikoba Ltd will be in control of the entire network. As soon as possible afterwards, Sikoba Ltd will create the Sikoba Foundation and seek to transfer operational control of the network to the Foundation Federation. The goal is for the Sikoba Network to become a decentralised, self-organising entity.

7.1 The Sikoba Foundation

The Sikoba Foundation will be a non-profit organization, registered in a suitable jurisdiction. Its role will be to act as a caretaker of the Sikoba network, including funding any necessary software development and projects that are useful for the Sikoba ecosystem. To fund its operations, the Sikoba Foundation will receive part of the transaction fee revenue generated by the Sikoba Network.

The Sikoba Foundation will be governed as a cooperative, with Sikoba token holders as well as Sikoba Federation members playing key roles in its governance. The precise governance model remains to be determined.

One of the main roles of the Sikoba Foundation will be to organise and lead the Sikoba Federation, and play a decisive role in the process of adding and removing Federation members.

7.2 The Sikoba Federation

The Sikoba Federation will be the collection of organisations and institutions who will be responsible for running the Sikoba blockchain nodes. Strict conditions will apply to become a Federation member, including a monetary deposit, the signing of a legal agreement with the Sikoba Foundation as well as the acceptance of terms and conditions that protect the integrity and privacy of the Sikoba Network.

The Sikoba blockchain will be a federated blockchain, in which only authorised nodes participate in the consensus protocol. These nodes will be run by Sikoba Federation members: highly trusted companies, organisations and institution. The Sikoba federation will maintain the blockchain, enforcing its economic rules. The user trust factor in this case is in the choice of the Federation members. No one member can cheat the system on its own, and as long as a large enough percentage of the members are honest, the blockchain remains intact.

The Sikoba Federation will be managed by the Sikoba Foundation, although it is expected that the Foundation will delegate some of its decision authority to the Federation members. It is also likely that there will be a significant overlap between Foundation stakeholders and Federaton members.

Chapter 8

Project status

The Sikoba Network is currently undergoing beta testing with the core backend and mobile application, while the blockchain layer and the web interface are still being developed.

8.1 Technology stack

The Sikoba core system is written in Crystal, a modern compiled language inspired by Ruby and built on LLVM. Data is stored in a PostgreSQL database. Status: beta. Codebase: 50,000 lines of Crystal.

For the blockchain/consensus layer, we are using Babble, a very simple blockchain middleware that provides consensus and ordering of transactions into blocks. Status: in development.

The mobile app called *sikobaPay*, is written in React Native and is available for both Android and iOS. Codebase: 35,000 lines of JavaScript. Status: beta.

The web app is written in ReactJS. Codebase: 8,500 lines. Status: pre-beta testing.

8.2 Mobile app

The *sikobaPay* mobile app was released in April of 2020. Together with the app release, the backend was also deployed on a single server. Although currently centralized, this setup allows for extensive testing to be done on the system and for users to get an initial idea of what the platform

is all about. Currently in its 0.4.4 beta version, the application is available for both Android and iOS devices.

Figures 8.1a, 8.1b, 8.1c and 8.1d show some of the most important screens in the app.

8.3 Current partnerships

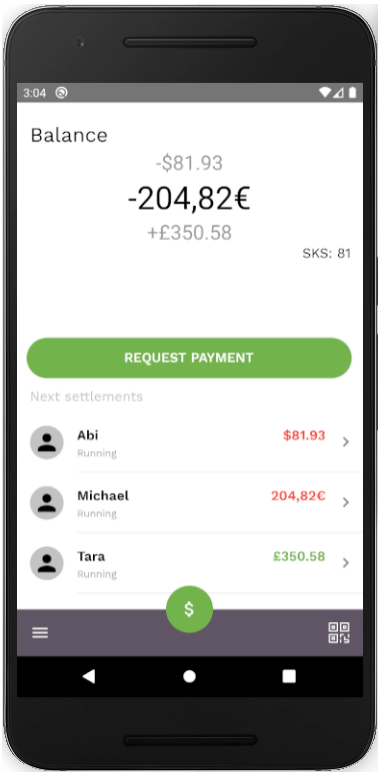
Sikoba is partnering with several organisations throughout the world to deploy the system and provide our software for various use cases. With this, both sides are getting a benefit. For our partners, we provide our versatile solution to solve their problems and satisfy their requirements. For Sikoba, we are getting right in the middle of already existing user groups and are leveraging this to get more users more efficiently and more quickly. We have several partnerships in the works, which are listed below.

8.3.1 Beki - Local Currency project in Luxembourg

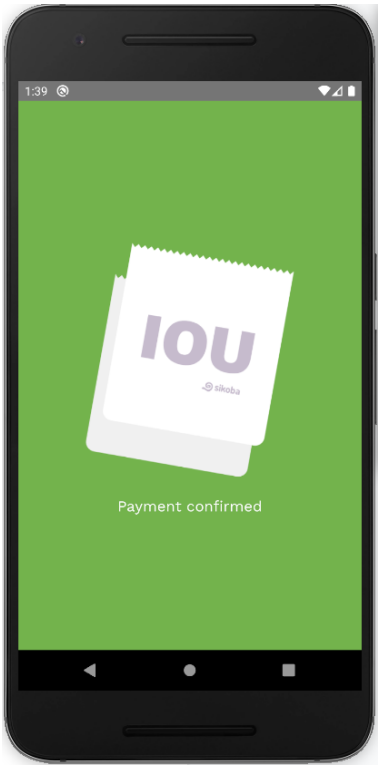
Sikoba has agreed with De Kär, operators of the Beki regional currency, to integrate the digital Beki into its platform.

De Kär is a non-profit organisation that manages the Beki regional currency in the canton of Redange, in the West of Luxembourg. The Beki is backed by the euro on a 1–1 basis and currently exists only in the form of paper currency, with denominations of 1, 2, 5, 10, 20 and 50 Beki. More than 100 local companies accept payments in Beki, and there are over 200,000 in circulation. Consumers can exchange 100 euros for 103 Beki, in effect giving them a 3% increase in purchasing power when buying locally.

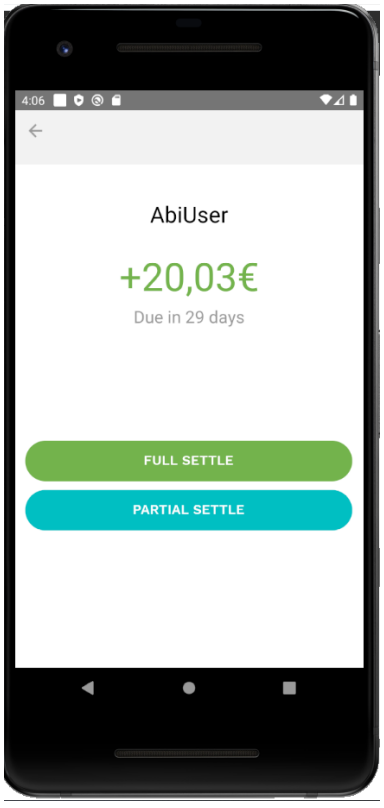
De Kär will work with Sikoba to introduce an electronic version of the Beki, the “e-Beki”, that can be used via the sikobaPay mobile app. This will reduce printing costs as well as eliminate the need to handle and count paper notes. In addition, local businesses will benefit from the IOU features of sikobaPay.



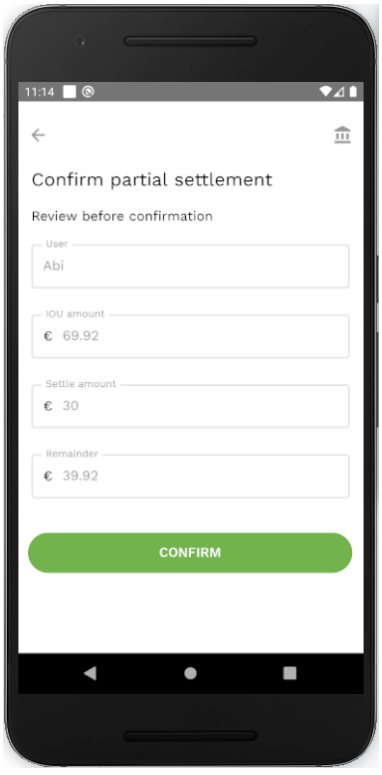
(A) sikobaPay - dashboard



(B) sikobaPay - payment



(C) sikobaPay - settlement



(D) sikobaPay - partial settlement

8.3.2 TeFía - Nano-Credit project in Bogotá, Colombia

Sikoba is partnering with Bogota-based Libranzas Group SAS to bring easy-to-use nano-credits to consumers in Colombia via a co-branded version of the sikobaPay mobile app, under the "Te Fía" brand. Te fía means I trust you in Spanish.

Instead of borrowing cash, Te Fía users will be granted credit lines which they can use in participating stores. Cash will be used only to reimburse outstanding balances, which can be done in any participating store. According to the Colombian Chamber of Commerce, around 120,000 small stores are registered in Bogotá alone, with an additional 230,000 stores in the rest of Colombia.

Te Fía will be launched as a pilot program in one Bogotá neighbourhood, but the goal is to expand to eventually cover the entire city of Bogotá, and later other regions of Colombia. The project is aiming to reach 1 million users within 12-18 months.

8.3.3 Jala - Community Credit in Manila, Philippines

Sikoba has entered into a partnership agreement with Jala, developers of the Jala ID+ Card, to help establish a local credit cooperative in the Payatas district of Quezon City, Manila.

For the past year, Jala has been working with the "Concerned Citizens of Payatas" initiative to establish a local credit cooperative. Payatas is a poor, slum-like district of Quezon City, which is the largest city in the Philippines and part of the Manila metropolitan region. Many inhabitants and small businesses in Payatas are exploited by loan sharks who charge interest rates of 150% per annum and more. The cooperative aims to provide community-funded loans with much lower interest rates.

Jala and Sikoba will work together to set up a pilot program with an initial 1,000 participants. Jala has the necessary local contacts, and will provide the Jala ID+ card to the program participants. Sikoba in turn will make its sikobaPay mobile app "Jala compliant", meaning that it will be able to communicate with Jala's ID+ card via NFC (near-field communication). The card will be used as a confirmation of a user's identity.

If the pilot project is successful, the goal is to expand the project not just in Payatas, but also in other districts of Quezon City and eventually in other cities in the Philippines. The goal is to reach at least 100,000 users within 12 months.

Bibliography

[1] Asli Demirgüç-Kunt Leora Klapper Dorothe Singer Saniya Ansar Jake Hess. Measuring financial inclusion and the fintech revolution. Technical report, World Bank Group, 2017.

[2] In the shadows. *The Economist*, March 2015. URL http://www.economist.com/node/2766310. last access 2020-01-06.

[3] URL https://ilo.org/global/about-the-ilo/newsroom/news/WCMS_627189/lang--en/index.htm. last access 2019-11-29.

[4] Alfred Mitchell-Innes. What is money? *The Banking Law Journal*, May 1913. URL https://www.newmoneyhub.com/www/money/mitchell-innes/what-is-money.html. last access 2020-08-20.

[5] Lars Boerner John William Hatfield. The design of debt clearing markets. Technical report, April 2016. URL https://static1.squarespace.com/static/551c608ce4b0f5016a1edc0f/t/572612f837013b4a807498b8/1462113017327/Debt+Clearing+Mechanisms+20161009.pdf. last access 2019-12-27.

[6] Lars Boerner John William Hatfield. The economics of debt clearing mechanics, May 2010. URL https://eh.net/eha/wp-content/uploads/2013/11/boerner.pdf. last access 2020-01-06.

[7] . URL https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki. last access 2020-08-17.

[8] Alex Biryukov, Daniel Dinu, and Dmitry Khovratovich. Argon2: New generation of memory-hard functions for password hashing and other applications. In *IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21-24, 2016*, pages 292–302. IEEE, 2016. doi: 10.1109/EuroSP.2016.31. URL https://doi.org/10.1109/EuroSP.2016.31.

[9] . URL `https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki`. last access 2020-08-17.