DEXTF

decentralized traded funds

Final Report **Jan 2020**

Author: Memento Blockchain Pte Ltd

Table of Contents

Executive Summary

In this proposal, we are delighted to present the DEXTF protocol, a secure, scalable and peer-to-peer solution for digital asset management on blockchain. The **focus of this POC is to develop a specific application of the DEXTF protocol to Family Offices which aims to directly connect clients owning cryptocurrencies with the investment team of Family Offices** (who are authorized to provide investment management services). Given the volatility and the information asymmetry associated with digital assets, and the potential for cyber-attacks, clients would find beneficial to use the services of professional investment managers to prevent the implementation of makeshift strategies, which often contribute significantly to destabilize markets, as exemplified by the breakneck speed at which value was created and destroyed in cryptocurrencies in 2018 alone.

Based on our industry experience in traditional and digital asset management, there are four fundamental challenges in the value chain of asset management.

- **Custody**
- **Rebalancing**
- **Pricing**
- **Execution**

This POC aims at addressing the first (Custody) of the four fundamental challenges. We believe this is the cornerstone of an infrastructure which can support a thriving digital asset management ecosystem. In our vision of the future, we believe that the full and complete infrastructure that will emerge after addressing all 4 challenges will enable a robust and secure development of the digital asset management industry, reduce infrastructure cost for digital asset managers, and increase transparency and returns for investors.

Our objective is for DEXTF to become the first solution that enables the transition from a centralized & traditional management of tokenized assets, to a decentralized & cryptographically secured management of digital assets. This will enable professional asset managers to focus on managing investment and asset allocation, while relying on a decentralized infrastructure to manage the custody of digital assets.

The speed at which digital assets are gaining popularity both with investors and family offices have simply made extremely important to create a platform that connects individual investors with professional portfolio managers.

Overview of the POC

The POC focuses on the use of Smart-Contracts, and applying it to the concept of Discretionary/Advisory Portfolio Management, taking into account that investors will want to have control over the digital tokens/assets that they own.

The POC has 2 use-cases,

- **Discretionary Portfolio Management**, where a portfolio manager has full investment decision-making over the investor's asset, and can manage the portfolio within certain agreed mandates
- **Advisory Portfolio Management**, where a portfolio manager does not have investment decision-making over the investor's asset, and has to seek client's permission before executing trades/re-balancing

The experiment is to determine if Distributed Ledger Technology (DLT) and cryptography can influence/affect the balance and control of digital assets between investors and portfolio manager for these 2 use cases, without a centralized custodian.

Specifically, it is to **consider different quorum of Multi-Signature Smart-Contract**. A x-of-y Multi-Signature Smart-Contract means there are a total of y keys that can trigger the Smart-Contract, only x keys are needed to access/initiate the Smart-Contract. Implicitly, x is smaller than y.

DEXTF can have a transformative impact in enabling a thriving and safe digital asset management industry. Our company is incorporated in Singapore, and our core team is fully based in Singapore; we believe this can also create a profound opportunity to position Singapore as a leading center for digital asset management which, in our view, will scale to become multiple times bigger than the current asset management industry.

Our Proposed Solution

Based on our industry expertise and on multiple discussions with various market participants, **we have identified 4 elements that are critical (necessary but not independently sufficient)** to address in a solution that wants to provide a robust infrastructure to enable the growth of a safe digital asset management sector:

1. **Custody**: the underlying blockchain technology must enable the client to retain full control of the assets without intermediaries, while enabling the Portfolio Manager to operate on them;
2. **Rebalancing**: the structure must allow Portfolio Managers to re-balance portfolios without giving them full control over the underlying assets;
3. **Pricing**: we need to deal with the problem of pricing of both the underlying and the fund;
4. **Execution**: we need a platform that allows cheap and real-time subscription, redemption and transferability.

For a solution to be successful, all four requirements need to be addressed simultaneously (vs tackling only some of them).

Presently, having reviewed all the existing blockchain and non-blockchain projects, there is no solution that addresses all the four challenges we identified above, positioning DEXTF at the forefront of this opportunity.

Our holistic approach tackles all 4 issues related to the rebalancing, custodizing, pricing and execution. DEXTF maintains complexity in place by clarifying from the outset the basic elements required to build a truly generalist digital asset management infrastructure.

To this end, we leveraged blockchain technology in combination with our significant industry expertise in traditional Asset Management. With DEXTF, Investors are bridged peer-to-peer with Portfolio Managers (PM), eliminating the need for PMs to custodize the clients' private keys. The protocol enables Family Offices (and other Portfolio Managers) to expand their client offering and market their expertise to clients. The benefits for the clients are conspicuous allowing them to enable professional PMs to manage their portfolio without losing ownership and control of their assets. Furthermore it makes the PMs more accountable for the performance of the portfolio and transparent with fees, and provides a solution which is based on cryptographically-secure principles, guarding against human fallacy by operations staff or digital manipulation by malicious actors. By tackling the custody problem, investors will be rewarded with transparency as to where exactly the fees are being spent as well as appraise the portfolio managers fairly and efficiently through distributed transactional records.

We have seen strong interest from Family Offices for our solution, and have **partnered with Mindful Wealth**, a Singapore-based Family office, to conduct the POC, as well as the pilot.

The DEXTF protocol provides a fully functioning platform on the Ethereum blockchain to connect Family Offices and their clients in a peer-to-peer manner, allowing the Portfolio Management team of the Family Offices to manage the digital assets of the Principal, without owning the private key of the assets. The Principal will continue to control the private keys at all times and hence will be able to choose the best custody solution according to his risk tolerance. The Portfolio manager never touches / owns the Clients' private keys and can focus on providing the optimal portfolio allocation based on the client's risk profile.

The DEXTF protocol involves the following elements:

1. **Portfolio Managers**
2. **Client**
3. **Basket of digital assets**

There are different implementation 'structures' for the portfolio management that will be considered. We devised a solution where the funds are completely tokenized so that the investors hold the keys of the token funds. The original digital assets instead are stored by smart custodian, i.e. a smart contract that uses a *Omnibus Model* for custody. In this way the problem is simplified and we just need to keep track of who is holding the private key for the maintenance of the smart custodian.

See section **The smart custody problem: who owns the private keys?** for a discussion on who should hold the administrator keys that control the funds of the smart custodian.

The choice of the blockchain and the smart contracts

In this section we describe and motivate the choice of the blockchain and the structure of smart contracts that allows us to create the framework.

Choice of the appropriate distributed ledger technology

The world of distributed ledger technologies (DLTs for short) is becoming increasingly complex as more and more projects claim to be the best infrastructure that there is. However most projects make claims on what will be delivered in the future failing to clearly state what tools are available right now.
Therefore, we spent significant time carefully analysing many different DLTs in order to choose the most appropriate one for our project. Among the numerous distributed-ledger projects available we short-listed the following two as possible platforms over which we could have built our solution:

1. The **Ethereum blockchain**, well known by developers and offering (almost) Turing-complete smart contracts.
2. The **Corda platform**: a distributed-ledger platform that allows certified peer-to-peer transactions. This solution also allows for (fully) Turing-complete smart contracts.

Since the research and development on the Corda platform initially requires significantly more resources than that on the Ethereum blockchain, we **decided to write the first version of this POC for the Ethereum blockchain**. This choice, made early in 2019, turned out to be fruitful as most projects about **Decentralized Finance (DeFi for short) decided to build on the Ethereum blockchain**.
Writing smart contracts on Ethereum, in principle, can be accomplished using a number of languages, however we found that the community has been growing around the **Solidity language**. Writing the smart contracts in this language allowed us to save significant time because of the large documentation on the subject already available on the internet.
Furthermore the smart contracts were developed on top of the **Truffle framework** that also allows the creation of unit tests to facilitate their maintenance.

There are other projects that we have considered for the purpose of this POC, however we deemed them not yet mature enough to be used in a production environment. These projects are:

1. the Cosmos universe (the internet of blockchains)
2. the Polkadot platform (united networks of state machines)
3. the Lightning network (scalable, instant bitcoin/blockchain transactions),
4. the Algorand blockchain (an innovative project that allows finality of blocks and simplified layer-1 smart contracts).

All the four above-mentioned projects had tremendous development during the year 2019. For example during the year 2019 the Algorand blockchain went from being a private net, to have a public test-net release, to have the main-net release; finally to release version 2.0 in November 2019. This incredible pace of evolution in such a short time, makes it even harder to choose *the best* DLT for the project.

In our first version of the software the user had to manually provide his private key in order to sign the transactions. As this could become a serious security problem we considered several other alternatives in order to protect the user private keys. The final choice was to **delegate all transaction signatures to MetaMask**. MetaMask is a bridge that allows the interaction of internet browsers with the blockchain. An interesting feature of MetaMask is that it also allows the use of hardware wallets in order to sign transactions (currently only the major hardware wallets, Ledger and Trezor, are supported). It also makes our solution more easily deployable by third parties as it delegates a key security risk (signing transaction) to an established, well documented and widely used solution.

Our software is composed of two main smart contracts: the ***multi-signature smart contract*** (MSC for short) and the ***custodian smart contracts***.

The multi-signature smart contract

Using the multi-signature contract it is possible to delegate the investment decision to multiple parties. For example, in a 2-out-of-3 multi-signature contract there is a total number of three users, each with his own private key, and it is necessary the signature of at least two users to validate any transaction. Since Ethereum, differently from other blockchains, does not have a native implementation of multi signatures one needs either to create his own version or to use an existing open-source solution.
On the Ethereum blockchain there are already a number of open-source versions of smart contracts that implement multi-signatures when signing blockchain transactions. In order to speed up development, and for security reasons, we choose to use an existing open-source multi-signature solution. Specifically we **decided to use the Gnosis Ethereum Multi-signature Wallet**, which greatly enhanced the flexibility of our multi-signature solution.

Using the Gnosis MS Wallet

The Gnosis MS Wallet features a well-tested user interface with open-source code: Ethereum Multisignature Wallet UI. This multi-signature wallet is very easy to use, for an example of basic usage you can watch this tutorial video. There is also a desktop interface that can be installed on major operating systems.

In the following paragraphs we give detailed instructions on how to use the multi-signature wallet for the most frequent operations:

- Create a new multi-signature wallet
- Create a multi-signature transaction
- Confirm a signed transaction
- Change the number of required signatures
- Add a new member or remove an old member

The flexibility of this multi-signature scheme allows scenarios where initially the power of signature is given to a certain number of users, however it can evolve to later include new users, or exclude existing ones.

Create a new multi-signature wallet

1. Go to the Wallet page
2. Click on the Add button on the top-right side
3. Enter the wallet name
4. Add, one by one, the public keys of the signers
5. Add the required number of signatures
6. Do not add a daily limit (it only works for Ether transactions)

Create a multi-signature transaction

1. Go to the Wallet page
2. Select the row with the Wallet you want to send money from
3. Select Withdraw on the far right, then choose the destination and the amount
4. Click the Send the mult-sig transaction button

Confirm a signed transaction

Suppose that you need to sign a transaction already created by another signer:

1. Go to the Wallet page
2. Select the row with the Wallet where the given transaction is supposed to be
3. Click on the Wallet name
4. Another window will pop up. Go to the bottom section, namely Multisig transactions
5. Choose the appropriate transaction

Change the number of required signatures

It is also possible to change the number of required signatures (or the daily limit):

1. Go to the Wallet page
2. Select the row with the Wallet you want to modify
3. Change the appropriate column
4. Ask the other signers to co-sign your transaction

Add a new member or remove an old member

Finally you can add and remove signers from a given wallet:

1. Go to the Wallet page
2. Select the row with the Wallet where the given transaction is supposed to be
3. Click on the Wallet name
4. Another window will pop up. Go to the Owners section
5. Add or remove signers
6. Ask for the required number of co-signatures

The ERC20 standard for assets on the Ethereum blockchain

In the early days of blockchain, following the example of Bitcoin, in order to create a new token you typically had to create a new Blockchain. The people using the flexibility of Ethereum smart contracts however realized that it was possible to use them to create new tokens. As more and more people created new tokens on the Ethereum blockchain, it became apparent that exchanging custom tokens was almost as hard as swapping token from different blockchains. In late 2015 a new standard was proposed: the ERC 20 standard. This new standard gave a common interface that could be used to create standardized tokens. The ERC20 standard specifies what methods must be implemented by the smart contract in order to operate as a token.

This standard was well received in terms of adoption by 2017 so much so that most Initial Coin Offering (ICO for short) were adopting it. By 2019 the movement of Decentralized Finance (DeFi for short) was evolving based on the principle of composability. Writing composable smart contracts means that their actions can be used by other contracts that follow the same standard. For example if a certain project builds a lending platform for ERC20 tokens, then it can be used by all projects representing assets in the ERC20 standard.

Hence, in order to maximise the possibility of success of the project, not only **we require that a digital fund can contain only ERC20 tokens, we also make the digital fund itself an ERC20 token.**

The smart custodian

As we have seen earlier, the multi-signature contract can be used to assign the responsibility to sign transactions to more than one actor. In this subsection we revise the main smart contracts, namely the custodian contracts, that allows the creation of fungible fund tokens. We use the plural custodian smart contracts because, for technical reasons, the custody is implemented by several smart contracts on the Ethereum blockchain. However all these smart contracts can be seen as exposing a single interface as described later.

From the outside the fund tokens expose the standard ERC-20 interface. This means that investors that hold a digital fund can transfer any quantity of it to another address as if the fund is a common fungible token on the Ethereum blockchain. The custodian contracts also have the responsibility to keep track of the ownership of the assets themselves. Hence the custodian contracts acts very similarly to an actual custodian company in the traditional financial world. Therefore in the following paragraph we will also refer to the custodian contracts as the smart custodian.

The fungibility problem

One of the biggest challenges that we faced in building a fund token that represents a whole portfolio of assets is the need for fungibility. This means that two fund **tokens held by two different investors should not be distinguishable**. Another issue is the so-called **additivity of tokens**, where several tokens can be split into two parts, not necessarily of the same size; each part can then be sent to two different addresses, which in turn will both send them to a third party; finally the third party can reconstruct exactly the original tokens.
In order to solve this problem, we created the smart custodian (implemented by a number of smart contracts) that acts as a depository. When the user sends his assets to the smart custodian, those tokens are registered under his name so that he can withdraw them at any time.

Example

Consider for example a user named Ada that has 8 Bitcoins[1] and 900 Ethers in her wallet. She currently does not have assets held by the smart custodian. Graphically we can represent this state as the following chart:

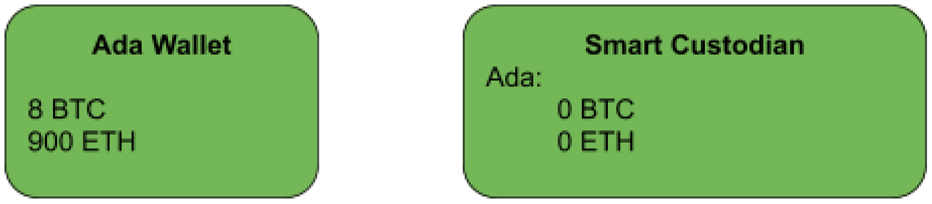


Figure 1

Since she plans to invest in a fund at a later date, she sends some of her tokens to the smart custodian. For example she sends 3 Bitcoins and 200 Ethers. At the end of the transaction the custodian contract will have registered these assets under her name, as shown in the following figure.

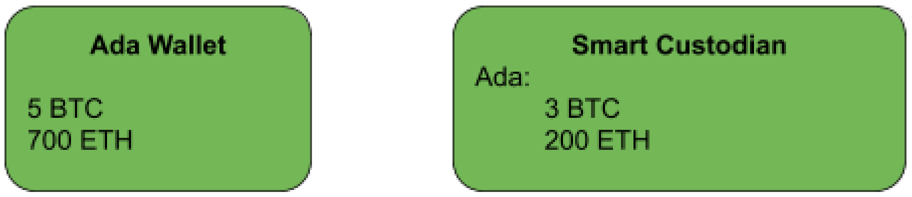


Figure 2

[1] Note that both Bitcoin and Ethers are technically not ERC20 tokens. However there are two wrapper contracts available, under the name of WBTC and WETH, that can be used as good proxies for Bitcoins and Ethers. In the rest of this paper we refer to these wrappers when using the names Bitcoin and Ether.

At this point let us consider a fund manager that wants to create a fund. He needs to decide an allocation, or base portfolio. For example he will create a base portfolio with 1 bitcoin and 50 ethereum.

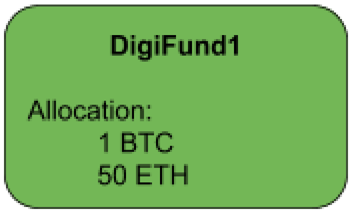


Figure 3

He can publish his allocation to the digital custodian so that anybody can convert 1 Bitcoin and 50 Ethers in one unit of the ***DigiFund1***.

Ada sees that a new fund is available and decides to *invest* in the fund 2 Bitcoins and 100 Ethers to receive 2 fund tokens. After Ada signs the transaction the blockchain custodian sends 2 DigiFund1 tokens to Ada and changes his internal bookkepeing by moving 2 Bitcoins and 100 Ethers from Ada's account to the DigiFund1 account.

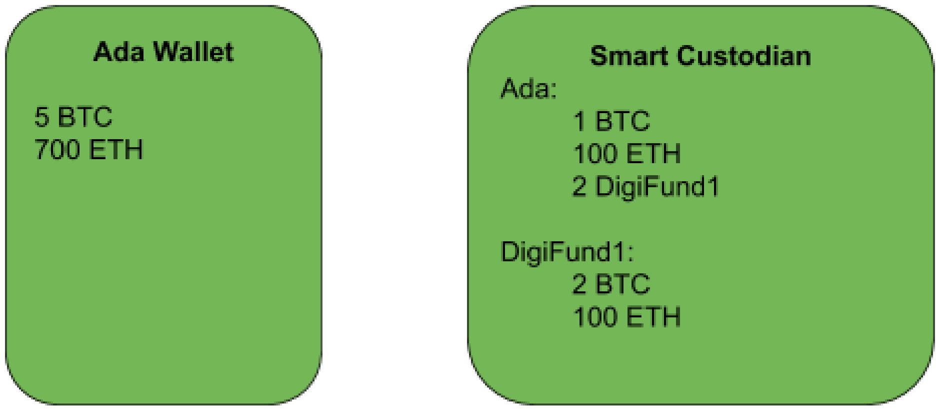


Figure 4

We notice that Ada has lost access to some of the assets, namely 2 BTC and 100 ETH, that now are registered under the ownership of DigiFund1. On the other hand Ada has gained access to 2 ***DigiFund1*** fund tokens and she can transfer them from the Smart Custodian account to her own account, as shown in the following graph

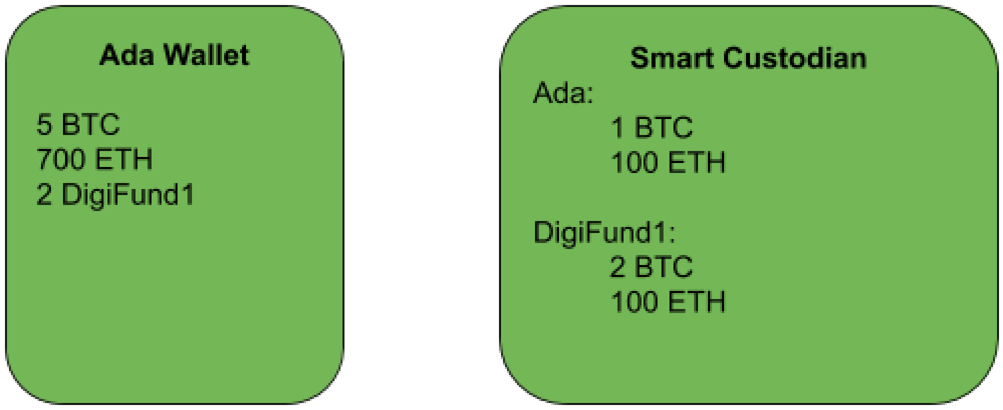


Figure 5

At this point Ada has the full ownership of 2 ***DigiFund1*** tokens, which are ERC20 tokens, in her wallet.
After some time Ada decides to send 1 of her DigiFund1 tokens to her friend Bob, that has no other tokens. After the transaction is mined on the blockchain Bob has one unit of DigiFund1. Notice that the smart custodian does not know who has access to digital funds, since the ownership of the fund token can change only by their owners.

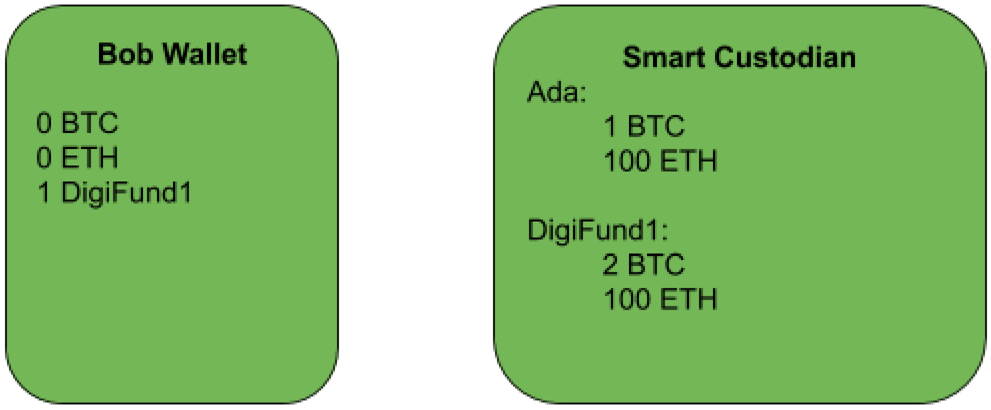


Figure 6

Bob now wants to redeem his token for the underlying assets, so he sends a redeem instruction to the blockchain custodian. After the transaction has been validated by the blockchain the state of assets are as follow (with Ada assets unchanged):

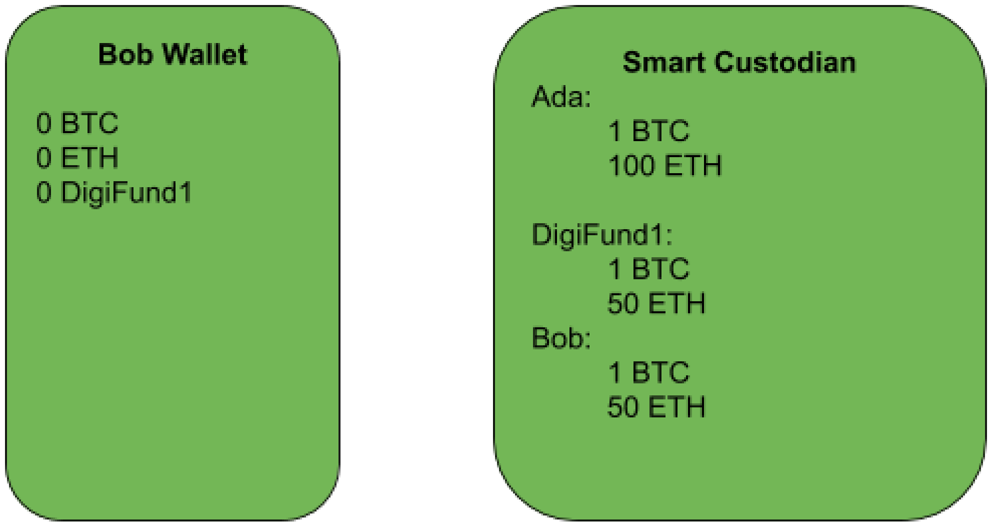


Figure 7

At this point, since there are assets in the custodian contract under the name of Bob he can send an instruction to have his assets moved to his own private wallet.

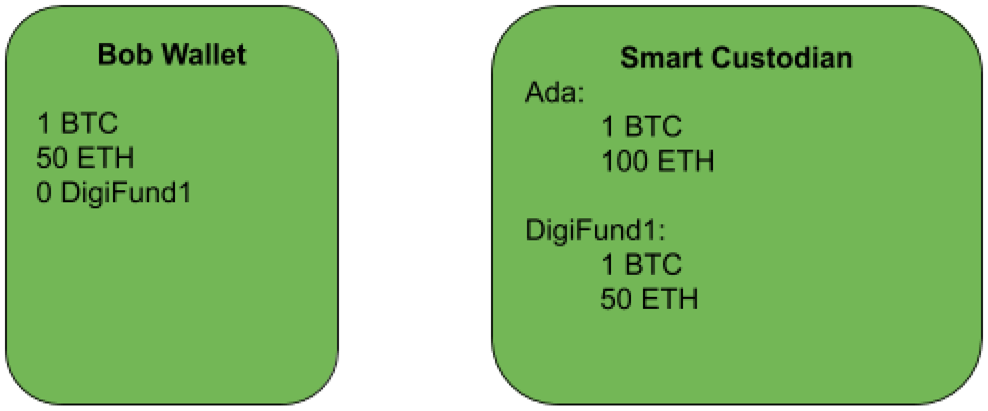


Figure 8

In summary, the custodian smart contracts allow for the creation and the destruction of fungible fund tokens simply by keeping track of who owns what tokens. This trick mainly works because the custodian contracts have the ability to generically assign assets to owners of fund tokens.

Summary of the custodian actions

The custodian smart contracts are described by computer code that can be executed by the blockchain nodes. In order to execute the different smart-contract methods a user needs to send the appropriate transaction to the blockchain. In our solutions there are three different type of **user roles**:

1. The **fund manager**, who is able to create digital funds by defining allocation of assets (base portfolios)
2. The **investors**, who have assets in their own wallets, or in their name under the smart custodian
3. The **administrator**, in charge of the maintenance of the custodian smart contracts

According to their role, different users can interact with the custodian smart contracts only on specific instructions

The fund manager role

The role of the fund manager is to create an allocation for other investors. Hence he can only perform one **operation**:

1. **Create a digital fund** by specifying how many units of different assets constitute one unit of the token fund

The investor role

The investor has control of his own assets and may want to have exposure to one or more digital funds. The operations he can perform are:

1. **Send his own assets to the smart custodian** so that they will be registered under his name
2. **Retrieve from the smart custodian the assets** that are registered in his name
3. **Instruct the smart custodian to take the appropriate allocation of assets** that are under his name, to register them under a certain digital fund, and finally to receive back in his wallet the corresponding number of units of the digital fund
4. **Redeem a digital fund** for its underlying assets

The administrator role

The administrator makes sure that everything runs smoothly in order to minimize the possible problems that the smart contract may present. The operations he can perform are:

1. **Freeze** the operations on a certain digital fund when it is appropriate
2. **Change the name under which the assets are registered** by the smart custodian
3. **Transfer funds** from the smart custodian to any external address

There are **different reasons for which we might need an administrator**:

- For **regulatory purposes** the administrator may be required to freeze or seize assets. This is for example in cases where there is a suspicious transaction or where the portfolio manager licence is suspended. Other typical requirements will be around AML where the administrator may be required to take actions on some investors’ assets.

- For **law-enforcement purposes** the administrator may be required to freeze or transfer the assets of some accounts. This is for instance required if the assets are stolen or in case a judge instructs the liquidation of an estate.
- For **security reasons**, for example when an hacker is somehow able to put some custodian funds under his name
- For **technical reasons**. Since at this point (early year 2020) all blockchains are still experimental, there might be a change of the behaviour of smart contracts that has not been foreseen by the designers

The smart custody problem: who owns the private keys?

In the previous section we described a possible solution to the problem of digital funds. In this solution the full custody of the fund assets is delegated to the custodian smart contracts that keeps track of the ownership of the assets themselves. However, the investors have access to the fund tokens in their portfolios. In this way the investors do not need to give up the ownership of the fund tokens. Also in this solution the fund manager only job is to dictate in which proportion the assets compose a digital fund, without having the possibility to gain access to the asset funds themselves. Finally investors and fund managers have distinct roles and have access to separated smart custodian functionalities.

Who owns the administrator keys?

While solving many problems regarding the digital funds themselves we have introduced another problem: **who owns the administrator keys?**
As we have seen the administrator has a lot of power as he can change ownership to the assets under custody. Therefore we believe that such a **power should be split to different parties using a multi-signature contract** as described earlier. In the following paragraphs we discuss different options for holding the administrator keys using multi signatures.

The investors hold one or more of the administrator keys

The investors already hold the private keys of their wallet using which they have both access to their digital-fund tokens and to the assets held by the smart custodian. We believe that **investors already have control over their own assets and should not be able to control the assets of other users**. Also as investing in a digital fund is the same as holding a fund token it is not known in advance who the investors are.
Investors are also protected because in our construct they can always destroy their digital funds and release the underlying assets which they will fully control.

The fund manager holds the keys

The fund manager job is to search for an appropriate allocation of digital assets as to satisfy a certain number of performance and risk requirements. As such the **fund managers should not have direct access to the investor's funds**. Therefore we believe that the fund manager should not hold the majority of the private keys of the administrator account. However, since somehow the fund manager is *responsible* for the fund itself he should also hold some power on the fund administration. Therefore we believe that the fund manager should hold at least one of the private keys that control the multi-signature account of the administrator.

External registered traditional custodians hold the keys

As the responsibility of a traditional asset-manager custodian is to ensure that the rightful owners of funds has access to his own assets, it is only fit that they could be involved in holding some of the administrator private keys. Actually the fund manager may seek the help of one or more custodians and give each of them a single key to the multi-signature contract that has administrator privileges.

Nobody holds the administrator keys

One could, in principle, also decide that fund administrators are not needed and **destroy completely the access to the administrator functions**. While this indeed could be an interesting choice that would actually completely put the investors under control of their own assets, we believe that the current state of blockchain is not mature enough for this choice. Furthermore, we believe that for the specific case of the Ethereum blockchain the development of the software is still not mature enough to destroy the administrator keys. For example after the deployment of the Istanbul fork, deployed in December 2019, a number of smart contracts stopped working. Currently we cannot be 100% sure that the deployed smart contracts on the Ethereum blockchain will continue to run as intended for a long period after they are deployed.

Solution Architecture

When designing the system, we committed to achieve the following key features:

- Modular
- Robust
- Secure
- Measurable
- Debuggable
- Maintainable
- Reusable
- Extensible

The layered (n-tier) high level architecture is depicted in *Figure 9*.

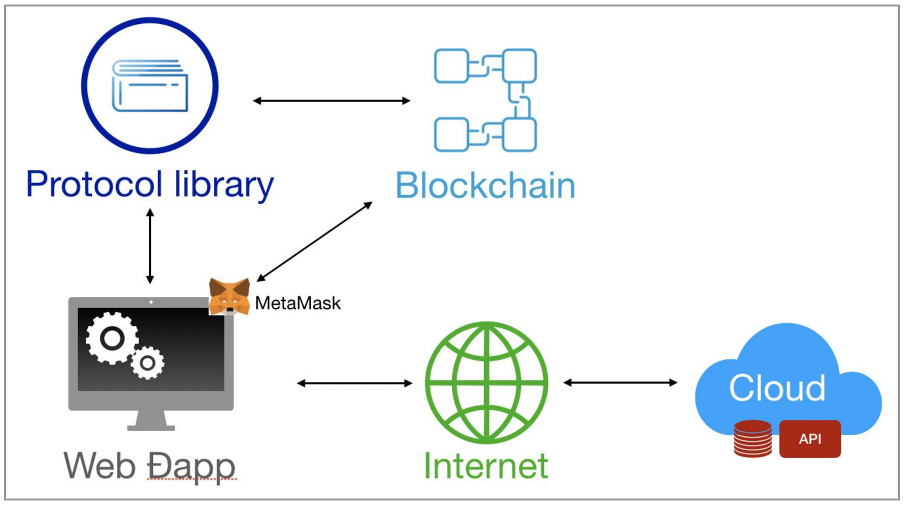


Figure 9: high level architecture

The key parts are:

Protocol library
A set of JS modules to manage the interactions with the blockchain (creation of a fund, issuance of a fund, etc.)

Web Ðapp
The front-end is a web-based interface, developed using **Vue.JS** (progressive framework for building user interfaces), **Semantic UI** (development framework that helps create responsive layouts) and the **MetaMask** add-on (bridge to the Ethereum blockchain)

Cloud services
Google Cloud Platform (GCP) services like **Firestore** (database) and **Cloud Functions** (API)

System UI

The front-end is a web-based app that can run on any browser, allowing the system to be **platform independent**.
We invested a considerable amount of time in designing the User Experience (UX) so that the actions of the various users (fund managers and investors) are as simple as possible.

The welcome page (*Figure 10*) invites the user to login (*Figure 11*) or register to the service.

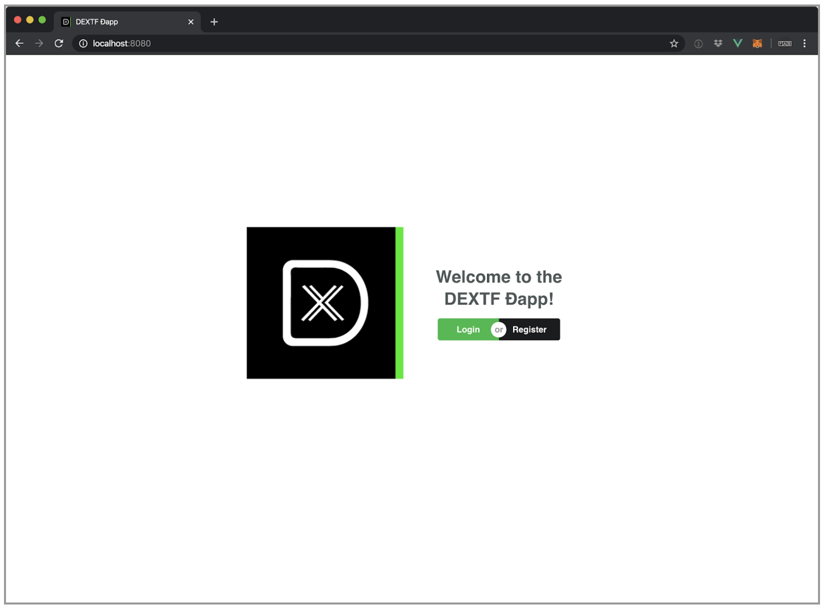


Figure 10: welcome page

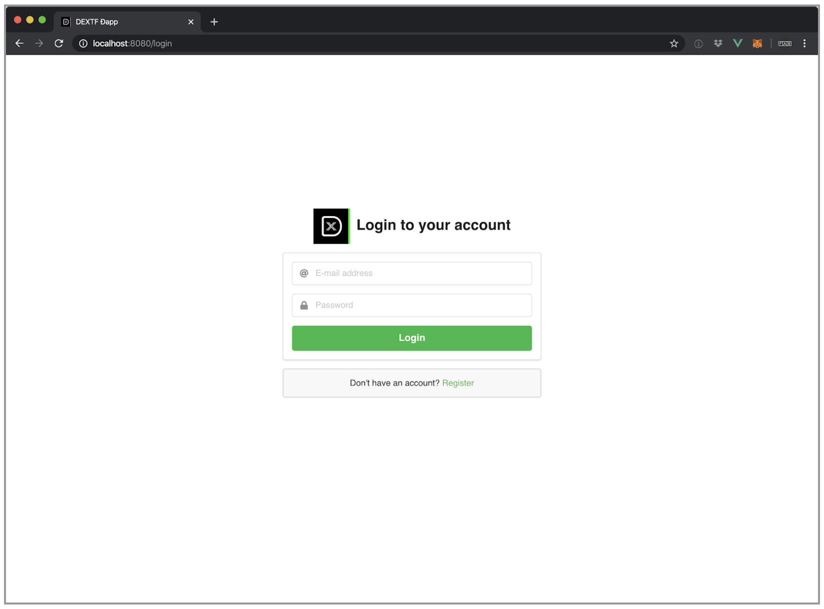
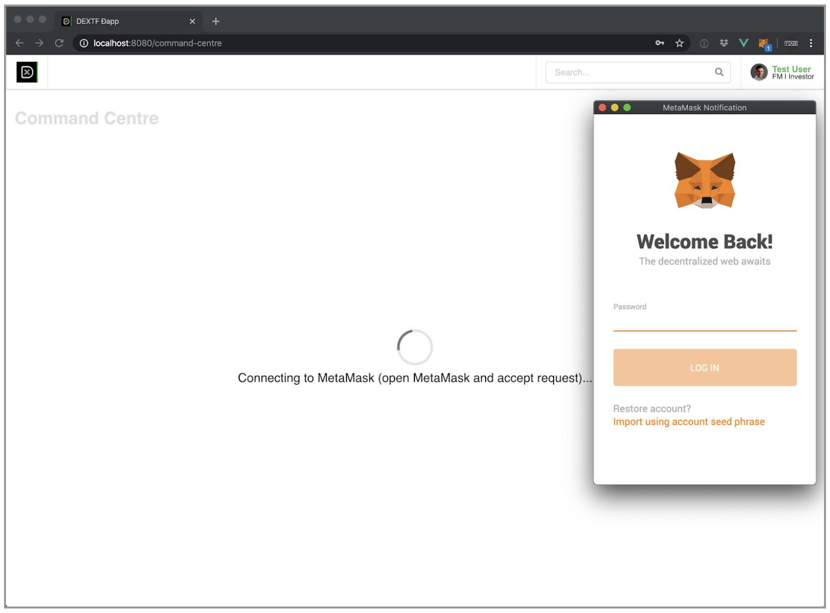


Figure 11: login page

In order to operate on the Ethereum blockchain the web browser uses the Metamask add-on. After successful authentication the user is invited to login to MetaMask and authorize the Đapp to connect to the user's wallet account (Figures 12 and 13).

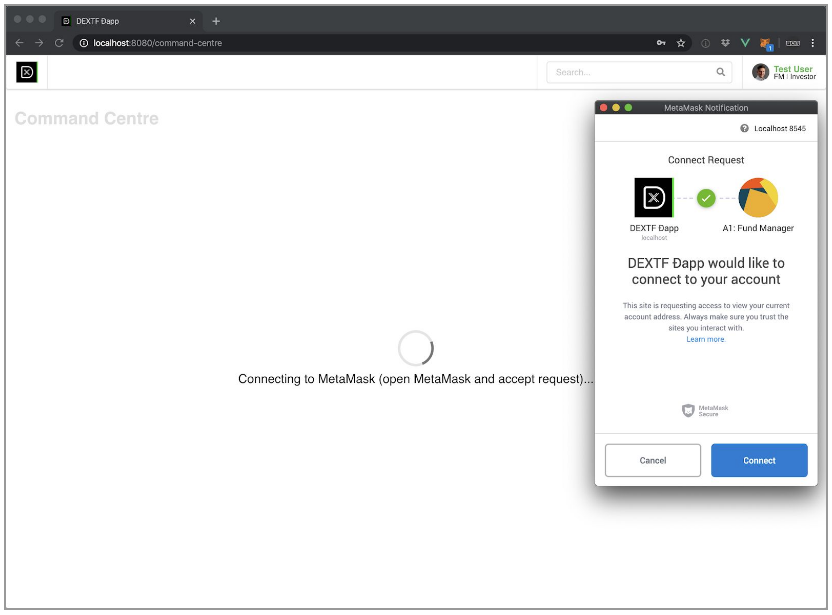


Welcome Back!

The decentralized web awaits

Connecting to MetaMask (open MetaMask and accept request)...

Figure 12: connect to MetaMask



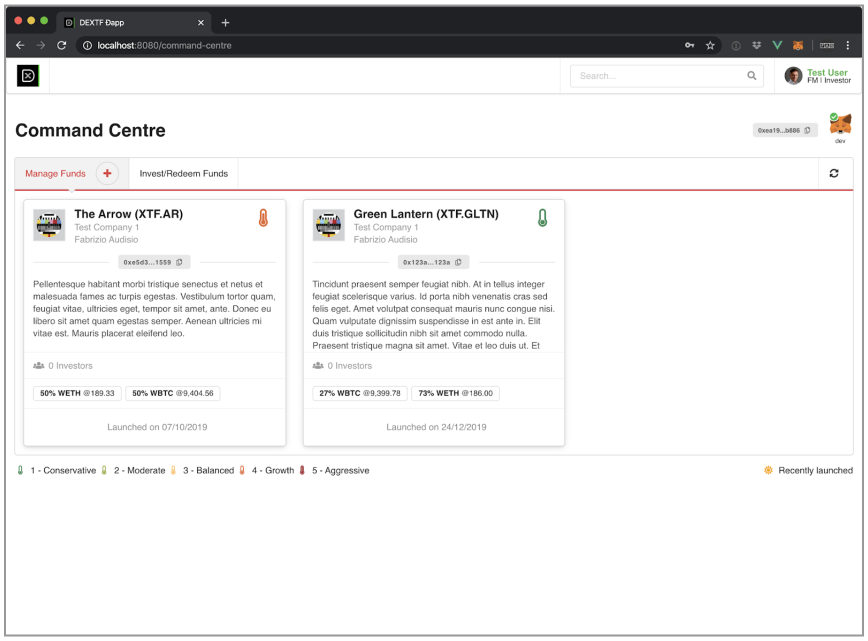
DEXTF Ðapp would like to connect to your account

This site is requesting access to view your current account address. Always make sure you trust the sites you interact with. Learn more.

Connecting to MetaMask (open MetaMask and accept request)...

Figure 13: authorize the Ðapp

The main page is the “Command Centre” (Figure 14). the “Manage Funds” tab is available to Fund Managers to manage their funds while the “Invest/Redeem Funds” tab is accessible by the Investors to invest into the publicly available funds.



The Arrow (XTF.AR)
Test Company 1
Fabrizio Audisio
0xe5d3...1559

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

0 Investors

50% WETH @189.33 | 50% WBTC @9,404.56

Launched on 07/10/2019

Green Lantern (XTF.GLTN)
Test Company 1
Fabrizio Audisio
0x123a...123a

Tincidunt praesent semper feugiat nibh. At in tellus integer feugiat scelerisque varius. Id porta nibh venenatis cras sed felis eget. Amet volutpat consequat mauris nunc congue nisi. Quam vulputate dignissim suspendisse in est ante in. Elit duis tristique sollicitudin nibh sit amet commodo nulla. Praesent tristique magna sit amet. Vitae et leo duis ut. Et

0 Investors

27% WBTC @9,399.78 | 73% WETH @186.00

Launched on 24/12/2019

1 - Conservative 2 - Moderate 3 - Balanced 4 - Growth 5 - Aggressive

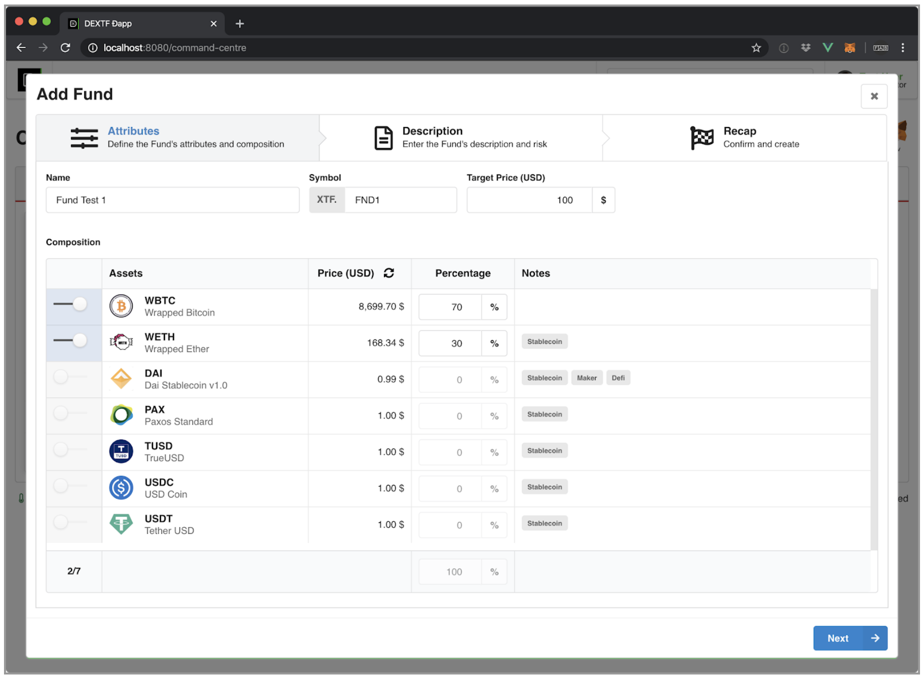
Recently launched

Figure 14: Command Centre - fund manager’s tab

Each fund is represented by a card with relevant information like name, issuing company, description, composition, etc…

Fund Manager UI

The “Manage Funds” tab shows all the funds that the Fund Manager created and manages. The ✚ button starts a step-by-step wizard to create a new fund (Figures 15, 16, 17).



DEXTF Dapp

localhost:8080/command-centre

Add Fund

Attributes — Define the Fund's attributes and composition

Description — Enter the Fund's description and risk

Recap — Confirm and create

Name: Fund Test 1

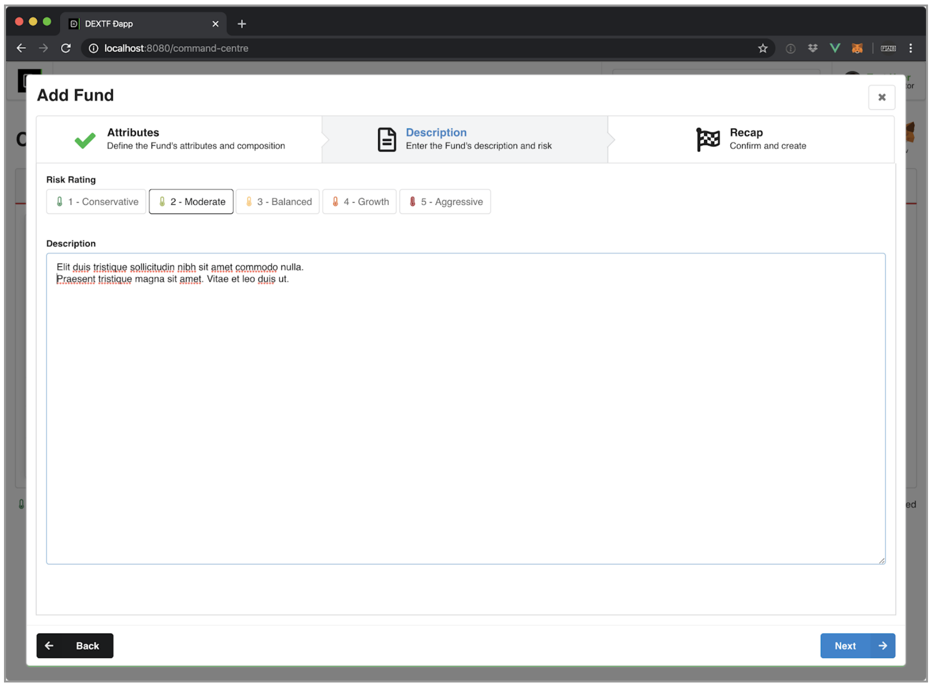
Symbol: XTF. FND1

Target Price (USD): 100 $

Composition

	Assets	Price (USD)	Percentage	Notes
	WBTC Wrapped Bitcoin	8,699.70 $	70 %	
	WETH Wrapped Ether	168.34 $	30 %	Stablecoin
	DAI Dai Stablecoin v1.0	0.99 $	0 %	Stablecoin Maker Defi
	PAX Paxos Standard	1.00 $	0 %	Stablecoin
	TUSD TrueUSD	1.00 $	0 %	Stablecoin
	USDC USD Coin	1.00 $	0 %	Stablecoin
	USDT Tether USD	1.00 $	0 %	Stablecoin
2/7			100 %	

Figure 15: fund attributes



DEXTF Dapp

localhost:8080/command-centre

Add Fund

Attributes — Define the Fund's attributes and composition

Description — Enter the Fund's description and risk

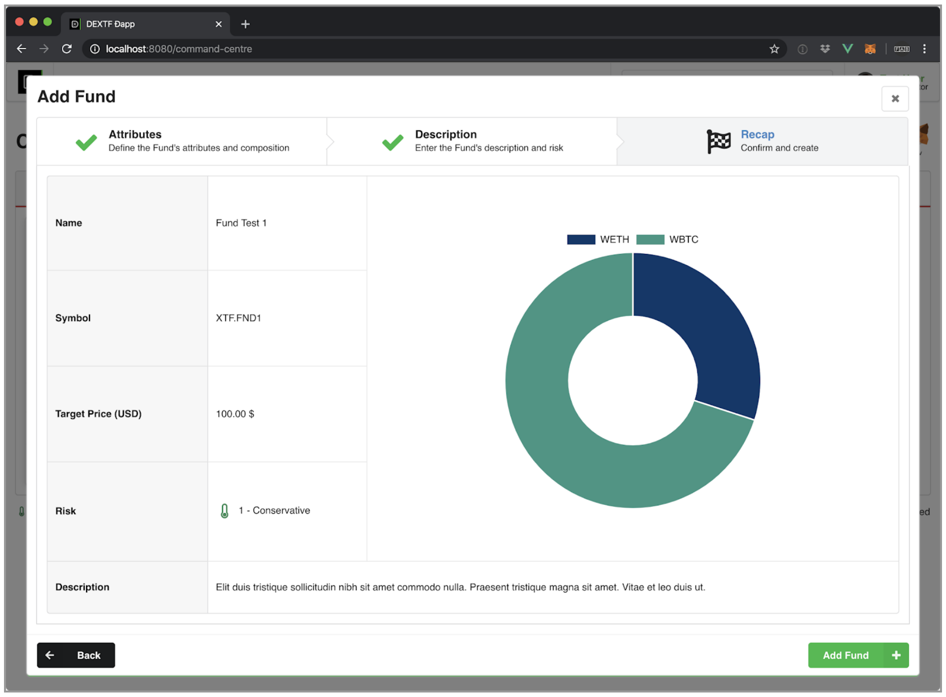
Recap — Confirm and create

Risk Rating: 1 - Conservative [✓ 2 - Moderate] 3 - Balanced 4 - Growth 5 - Aggressive

Description

Elit duis tristique sollicitudin nibh sit amet commodo nulla.
Praesent tristique magna sit amet. Vitae et leo duis ut.

Figure 16: fund description


DEXTF Dapp
localhost:8080/command-centre
Add Fund
Attributes
Define the Fund's attributes and composition
Description
Enter the Fund's description and risk
Recap
Confirm and create
WETH
WBTC
Back
Add Fund

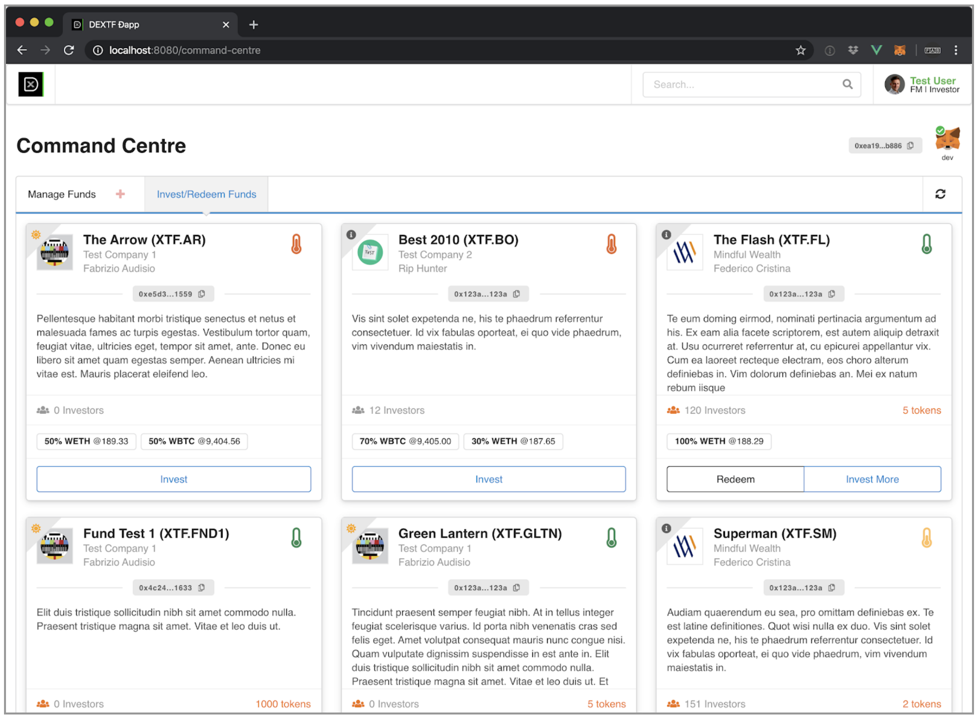
Name	Fund Test 1
Symbol	XTF.FND1
Target Price (USD)	100.00 $
Risk	1 - Conservative
Description	Elit duis tristique sollicitudin nibh sit amet commodo nulla. Praesent tristique magna sit amet. Vitae et leo duis ut.

Figure 17: recap

Once the Fund Manager is ready to launch the newly created fund he can press the **Launch** button (Figure 14); this will trigger a call to a specific function of the Protocol library and the creation of the smart contract representing the new fund.

Investor UI

The "Invest/Redeem Funds" tab shows all the publicly available funds (Figure 18). The Investor can perform two operations: Invest and Redeem.


DEXTF Dapp

localhost:8080/command-centre

Search...

Test User
FM | Investor

Command Centre

0xea19...b886

dev

Manage Funds | Invest/Redeem Funds

The Arrow (XTF.AR)
Test Company 1
Fabrizio Audisio
0xe5d3...1559
Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.
0 Investors
50% WETH @189.33 | 50% WBTC @9,404.56
Invest

Best 2010 (XTF.BO)
Test Company 2
Rip Hunter
0x123a...123a
Vis sint solet expetenda ne, his te phaedrum referrentur consectetuer. Id vix fabulas oporteat, ei quo vide phaedrum, vim vivendum maiestatis in.
12 Investors
70% WBTC @9,405.00 | 30% WETH @187.65
Invest

The Flash (XTF.FL)
Mindful Wealth
Federico Cristina
0x123a...123a
Te eum doming eirmod, nominati pertinacia argumentum ad his. Ex eam alia facete scriptorem, est autem aliquip detraxit at. Usu ocurreret referrentur at, cu epicurei appellantur vix. Cum ea laoreet recteque electram, eos choro alterum definiebas in. Vim dolorum definiebas an. Mei ex natum rebum iisque
120 Investors | 5 tokens
100% WETH @188.29
Redeem | Invest More

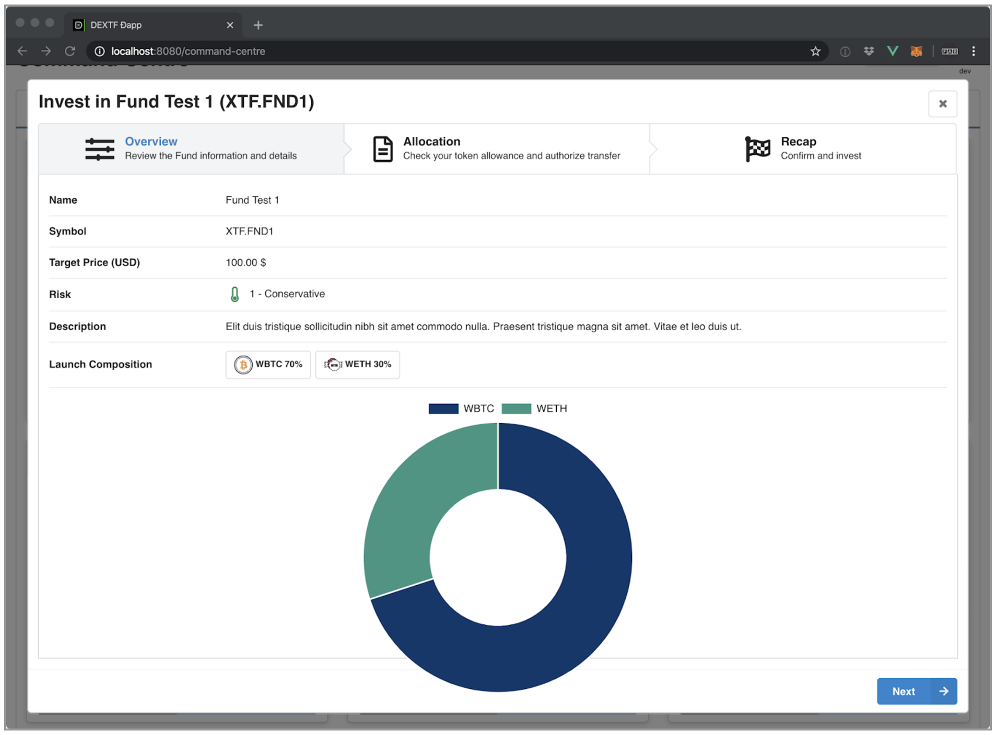
Fund Test 1 (XTF.FND1)
Test Company 1
Fabrizio Audisio
0x4c24...1633
Elit duis tristique sollicitudin nibh sit amet commodo nulla. Praesent tristique magna sit amet. Vitae et leo duis ut.
0 Investors | 1000 tokens

Green Lantern (XTF.GLTN)
Test Company 1
Fabrizio Audisio
0x123a...123a
Tincidunt praesent semper feugiat nibh. At in tellus integer feugiat scelerisque varius. Id porta nibh venenatis cras sed felis eget. Amet volutpat consequat mauris nunc congue nisi. Quam vulputate dignissim suspendisse in est ante in. Elit duis tristique sollicitudin nibh sit amet commodo nulla. Praesent tristique magna sit amet. Vitae et leo duis ut. Et
0 Investors | 5 tokens

Superman (XTF.SM)
Mindful Wealth
Federico Cristina
0x123a...123a
Audiam quaerendum eu sea, pro omittam definiebas ex. Te est latine definitiones. Quot wisi nulla ex duo. Vis sint solet expetenda ne, his te phaedrum referrentur consectetuer. Id vix fabulas oporteat, ei quo vide phaedrum, vim vivendum maiestatis in.
151 Investors | 2 tokens

Figure 18: Command Centre - investor's tab

The **Invest** button starts a step-by-step wizard to drive the users through the process to invest into a fund (Figures 19, 20, 21).


DEXTF Dapp

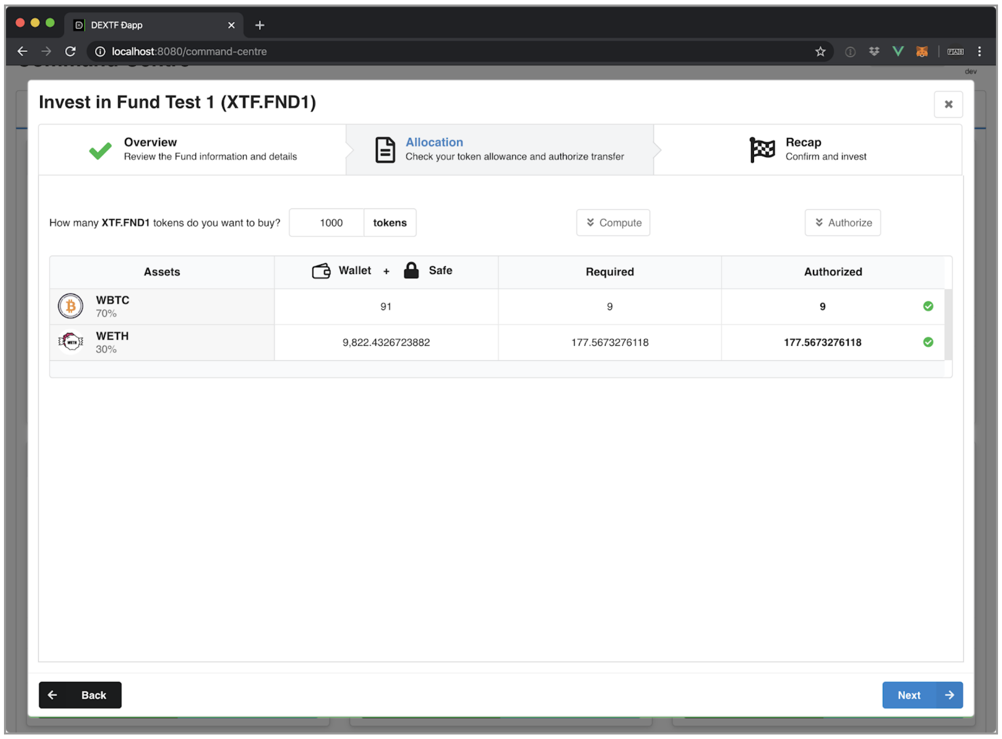
localhost:8080/command-centre

Invest in Fund Test 1 (XTF.FND1)

Overview — Review the Fund information and details | Allocation — Check your token allowance and authorize transfer | Recap — Confirm and invest

Name	Fund Test 1
Symbol	XTF.FND1
Target Price (USD)	100.00 $
Risk	1 - Conservative
Description	Elit duis tristique sollicitudin nibh sit amet commodo nulla. Praesent tristique magna sit amet. Vitae et leo duis ut.
Launch Composition	WBTC 70% WETH 30%

Figure 19: overview



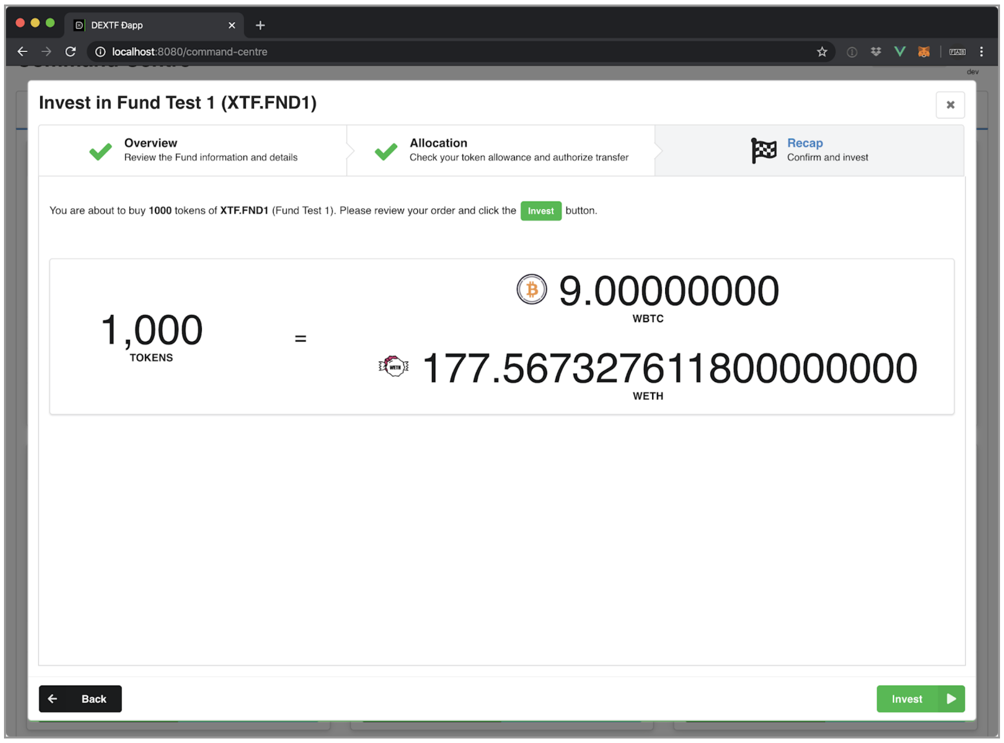
Invest in Fund Test 1 (XTF.FND1)

Overview
Review the Fund information and details
Allocation
Check your token allowance and authorize transfer
Recap
Confirm and invest

How many **XTF.FND1** tokens do you want to buy? 1000 **tokens**

Assets	Wallet + Safe	Required	Authorized
WBTC 70%	91	9	9 ✓
WETH 30%	9,822.4326723882	177.5673276118	177.5673276118 ✓

Figure 20: allocation



Invest in Fund Test 1 (XTF.FND1)

Overview
Review the Fund information and details
Allocation
Check your token allowance and authorize transfer
Recap
Confirm and invest

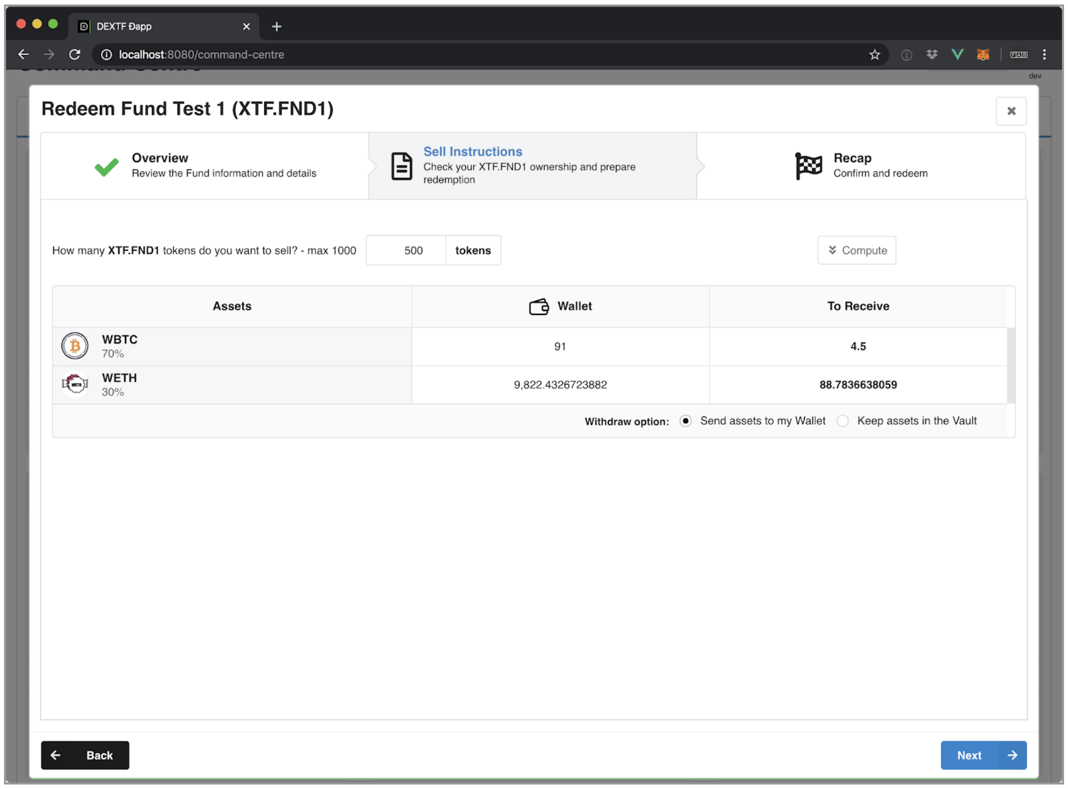
You are about to buy **1000** tokens of **XTF.FND1** (Fund Test 1). Please review your order and click the Invest button.

1,000 TOKENS = 9.00000000 WBTC 177.567327611800000000 WETH

Figure 21: recap

When the Investor presses the **Invest** button a call to a specific function of the Protocol library is triggered; this will start the blockchain transaction to complete the investment.

The reverse operation (redemption of the fund) is started with the **Redeem** button (Figure 18), that starts a step-by-step wizard to drive the users through the process to redeem a fund (Figures 22 and 21).



Redeem Fund Test 1 (XTF.FND1)

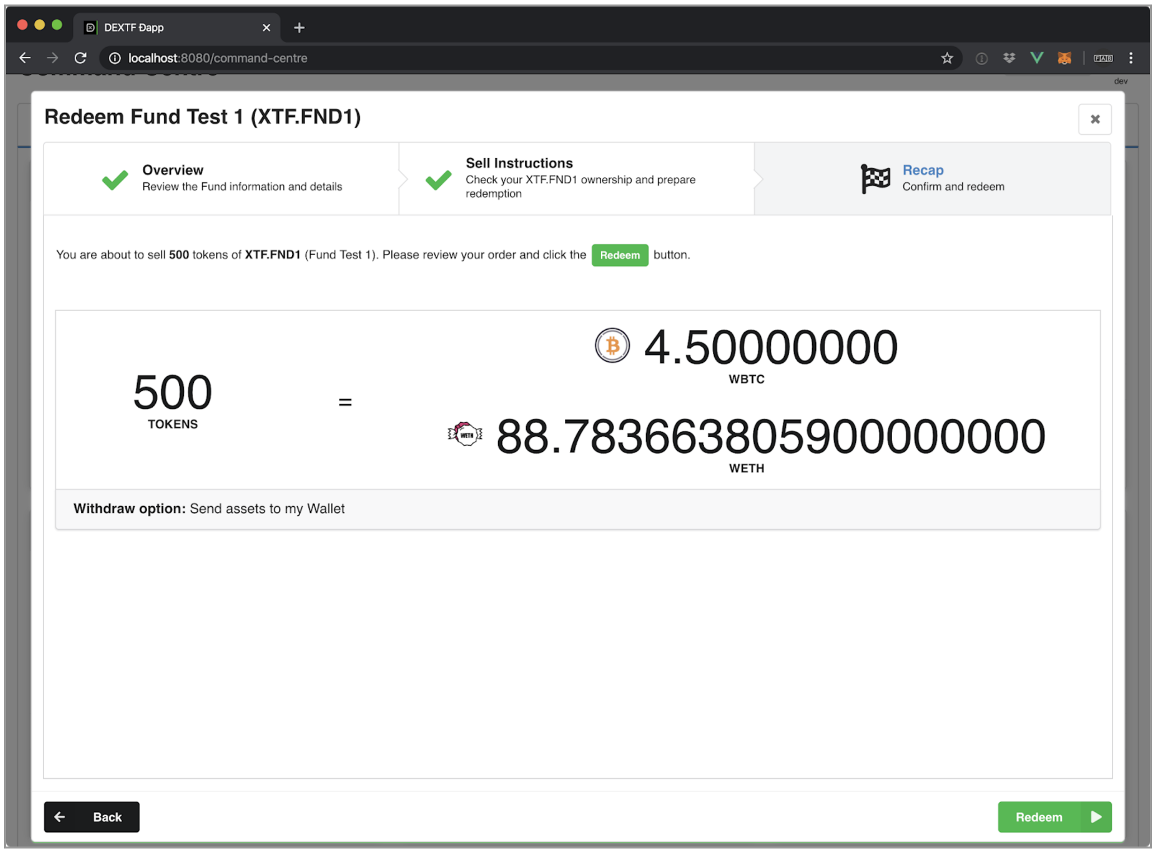
Overview – Review the Fund information and details
Sell Instructions – Check your XTF.FND1 ownership and prepare redemption
Recap – Confirm and redeem

How many **XTF.FND1** tokens do you want to sell? - max 1000: 500 tokens

Assets	Wallet	To Receive
WBTC 70%	91	**4.5**
WETH 30%	9,822.4326723882	**88.7836638059**

Withdraw option: [✓ Send assets to my Wallet] Keep assets in the Vault

Figure 22: redemption instructions



Redeem Fund Test 1 (XTF.FND1)

Overview — Review the Fund information and details

Sell Instructions — Check your XTF.FND1 ownership and prepare redemption

Recap — Confirm and redeem

You are about to sell **500** tokens of **XTF.FND1** (Fund Test 1). Please review your order and click the Redeem button.

500 TOKENS = 4.50000000 WBTC, 88.783663805900000000 WETH

Withdraw option: Send assets to my Wallet

Figure 23: recap

When the Investor presses the **Redeem** button a call to a specific function of the Protocol library is triggered; this will start the blockchain transaction to complete the redemption.

Limitations and future works

The structure of smart contracts described in this work allows the creation of funds where the fund manager defines the asset allocation and under no circumstances he will have control of the funds' tokens, whether they are representing the newly minted digital funds or the tokens representing the balance of each investor custodized by the Smart Custody contract. As a result, the completion of this POC enabled us to introduce and build a **working yet basic offering**, if compared to the complex financial products issued in the market today. With the current state of the protocol, we are able to issue structured notes, which are passive investments based on short-term strategies. Furthermore, we recognize that in order to realize the ideal but achievable goal of distinguishing Singapore as a trendsetter in the digital asset management industry, we need to extensively develop on top of the inefficiencies that corrode investors' value and stifle asset managers' performances within the traditional AM framework. To this end, we have outlined in our 2020 plan to develop and introduce a **key feature** which will enable our currently passive funds to **become active through rebalancing** instructed by the Portfolio Manager. In Portfolio Management, the possibility to rebalance helps portfolios to maintain a certain level of asset allocation, which is optimized based on risk profile taking into consideration asset diversification requirements among asset classes as well as hedging purposes. It is a feature that enables, on a digital platform, complex technical as well as financial decisions. We are also working on an in-depth market analysis to determine what could be an acceptable fee structure to introduce in order to ultimately launch a financially sustainable, safe and desirable (competitive) investment platform. For most part of 2020, we will work to involve market makers and exchanges (decentralized and non) for our rebalancing function to get access to best execution (best rates and minimal slippage).

We are extremely thankful for being awarded the MAS FSTI POC grant, and it has enabled us to complete this first backbone infrastructure that prepares us for our next challenge. We are also grateful for the feedback from the evaluation panel, who has taken time out to review the proposal for this proof of concept.