

Clear Contracts *Lightpaper*

An Introduction to Clear Contracts

Templated smart contract library DAO

[Join our Discord](#)

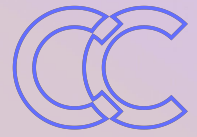


[Twitter](#)



[LinkedIn](#)





Abstract

Clear Contracts is a decentralized smart contract management architecture powered by the Cardano Blockchain. The architecture is carefully designed to provide a non technical solution for users to create and deploy automated decentralized executable code blocks. The platform enables users to create, manage, and execute custom smart contracts in a no-code manner.

The primary objective of Clear Contracts is to democratize the ability to interact with the Cardano blockchain by providing access to high assurance smart contracts through a no-code solution. Our contract solution will enable DAOs/dapps to easily implement smart contracts into their desired workflows and to create smart contract templates to facilitate DAO creation, governance, and execution.

Smart contracts are executable code blocks that automate transactions once specific conditions defined within the logic of their code are met. Smart contracts have the unique capability to automate transactions and track checkpoints without a trusted 3rd party.

Clear Contracts' platform facilitates the creation of custom smart contracts to allow individual or organizations to engage in peer-to-peer agreements.

Due to the capability of smart contracts to automatically verify stipulations and carry out transactions in conjunction with legal agreements, Clear Contracts' technology cuts out the need for middle men to be involved in the contract process.

Organizations, businesses, individuals, governments, and industries are becoming increasingly aware of how blockchain and smart contract technology can be implemented into their existing systems to give themselves a competitive advantage and cut costs.

The ability to provide a no code solution for any party to utilize this new technology to optimize processes will be needed as blockchain adoption continues to increase over the coming years.

This document lays out the technical foundations and economic mechanisms of the Clear Contracts architecture.

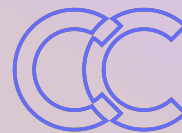
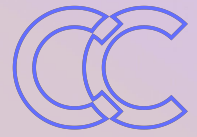


Table of Contents

Page	Section
1	Title
2	Abstract
3	Table of Contents
4	Our Mission
5	The Opportunity
6	The Solution
7	Technical Developments
8	Technical Developments
9	Roadmap
10	Tokenomics
11	Token Utility
12	Token Utility
13	Platform Architecture
14	Conclusion



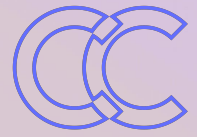
Our Mission

The mission of Clear Contracts is to lower the barrier to entry to create, manage, and execute smart contracts. Smart contracts have the ability to make existing systems more optimized and equitable.

Access to technology with this potential should not be limited to a small group of highly technical individuals. Rather, it should be accessible to as many people as possible.

Smart contracts are ready to disrupt existing systems based of their ability to utilize blockchain technology to track and automate transactions and processes in conjunction with legal documentation.

Clear Contracts is at the forefront of providing a non technical solution for this technology to be implemented into the industries who would benefit from it the most.



The Opportunity

The opportunity in front of us comes to light because of three facts of the blockchain industry.

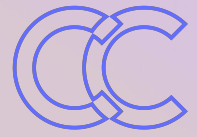
Writing high assurance smart contracts is difficult, time consuming, and expensive. It is extremely inefficient for each individual project in the Cardano ecosystem to devote time and money to develop much of the same code.



The technical barrier to entry to create and interact with smart contracts on the cardano blockchain is too high. Blockchain concepts like DAOs are not inclusive to all if only a select few individuals possess the technical ability to create and interact with them.

Obtaining contract audits is expensive and labor intensive. Small projects with the desire to create verifiably secure dapps have no affordable way to access audited contracts.





The Solution

Clear Contracts provides a non technical solution so that anyone can create, manage, and execute high assurance smart contracts.

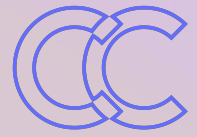
We provide a comprehensive solution that enables our users to interact with audited smart contract templates out of the box.

We turn complicated, audited haskell code...

```
1 {-# LANGUAGE TemplateHaskell #-}
2
3 module Week08.Lens where
4
5 import Control.Lens
6
7 newtype Company = Company {_staff :: [Person]} deriving Show
8
9 data Person = Person
10 { _name :: String
11 , _address :: Address
12 } deriving Show
13
14 newtype Address = Address {_city :: String} deriving Show
15
16 alejandro, lars :: Person
17 alejandro = Person
18 { _name = "Alejandro"
19 , _address = Address {_city = "Zacateca"}
20 }
21 lars = Person
22 { _name = "Lars"
23 , _address = Address {_city = "Regensburg"}
24 }
25
26 iohk :: Company
27 iohk = Company {_staff = [alejandro, lars]}
28
29 goTo :: String -> Company -> Company
30 goTo there c = c {_staff = map movePerson (_staff c)}
31   where
32     movePerson p = p {_address = (_address p) {_city = there}}
```

```
Withdraw (Wallet 1) (Wallet 2) 1 2
SetPrice (Wallet 1) (Wallet 2) 1
BuyTokens (Wallet 2) (Wallet 2) 0
Start (Wallet 2)
Start (Wallet 1)
AddTokens (Wallet 1) (Wallet 2) 6
BuyTokens (Wallet 1) (Wallet 1) 1
Prelude Plutus.Contract.Test.ContractModel Test.QuickCheck Spec.Model > :t nextState
nextState :: ContractModel state => Action state -> Spec state ()
Prelude Plutus.Contract.Test.ContractModel Test.QuickCheck Spec.Model > :t view
view
:: mtl-2.2.2:Control.Monad.Reader.Class.MonadReader s m =>
  Getting a s a -> m a
Prelude Plutus.Contract.Test.ContractModel Test.QuickCheck Control.Lens Spec.Model > :t perform
perform
:: ContractModel state =>
  HandleFun state
  -> ModelState state
  -> Action state
  -> Plutus.Trace.Emulator.EmulatorTrace ()
Prelude Plutus.Contract.Test.ContractModel Test.QuickCheck Control.Lens Spec.Model > :t propRunActions
propRunActions      propRunActions_      property
propRunActionsWithOptions properFraction propertyForAllShrinkShow
Prelude Plutus.Contract.Test.ContractModel Test.QuickCheck Control.Lens Spec.Model > :t propRunActionsWithOptions
propRunActionsWithOptions
:: ContractModel state =>
  Plutus.Contract.Test.CheckOptions
  -> [ContractInstanceSpec state]
```

into a simple online form.



Technical Developments:

Simple Escrow Smart Contract based off captured user interactions

One of our initial contracts is a simple escrow smart contract that takes the user's desired inputs, deploys a contract to the blockchain, and manages the eventual facilitation of funds to the correct party.

This contract can be applied across a variety of industries because it provides an easy way for any two parties to engage in a trustworthy and transparent agreement. In order to drive more adoption of this contract is capable of using any token including stablecoins, tokens for DAOs, ADA, etc...

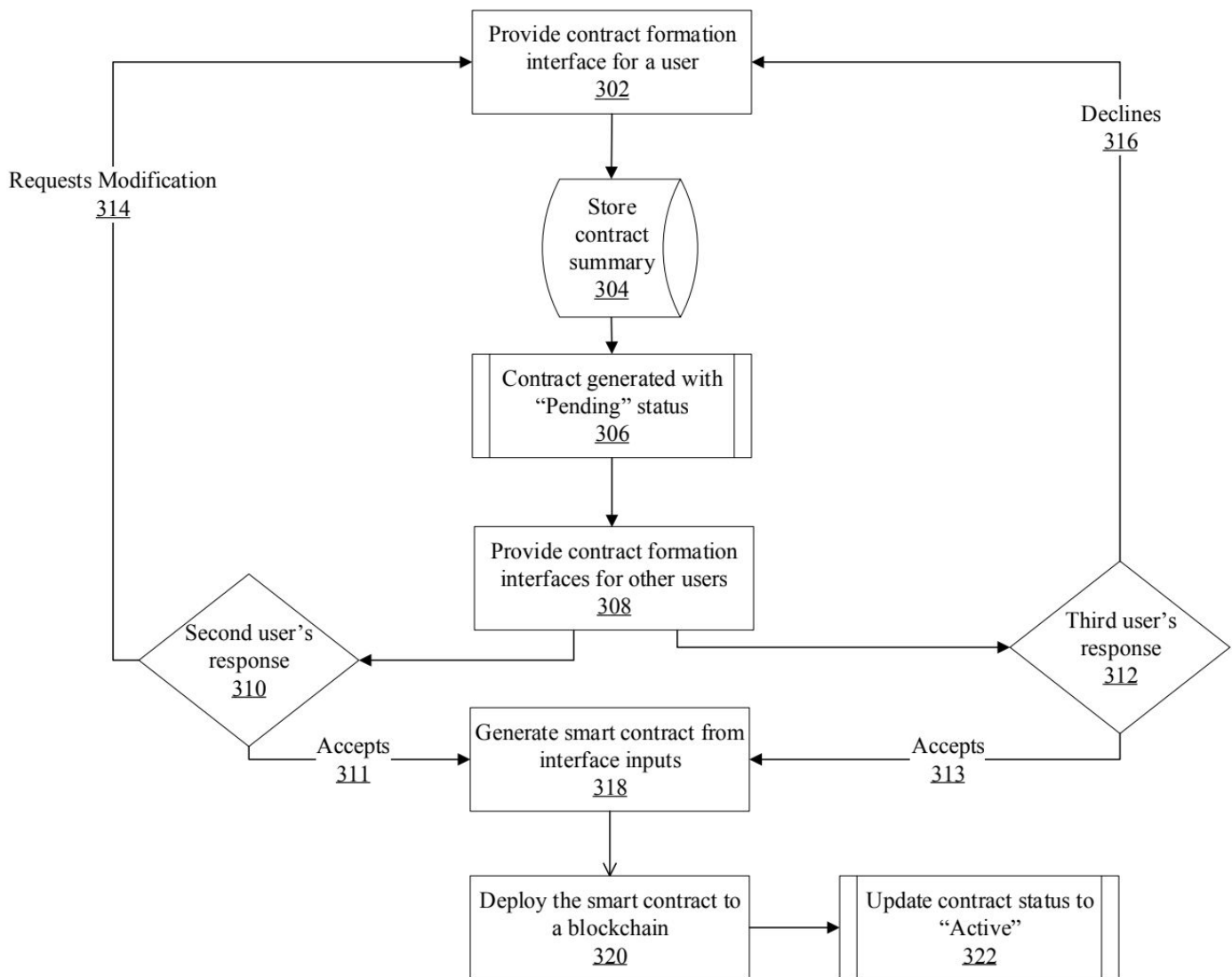
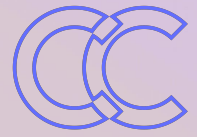


Fig. 1: Simple Escrow Smart Contract deployment logic



Technical Developments:

Simple Escrow Smart Contract based off captured user interactions Cont.

This flow chart details how the logic of the escrow contract ensures that funds are allocated to the correct party to complete execution of the contract based off captured user interactions. In the event of a disagreement between the parties, an arbitrator is assigned at the beginning of the process to resolve disputes.

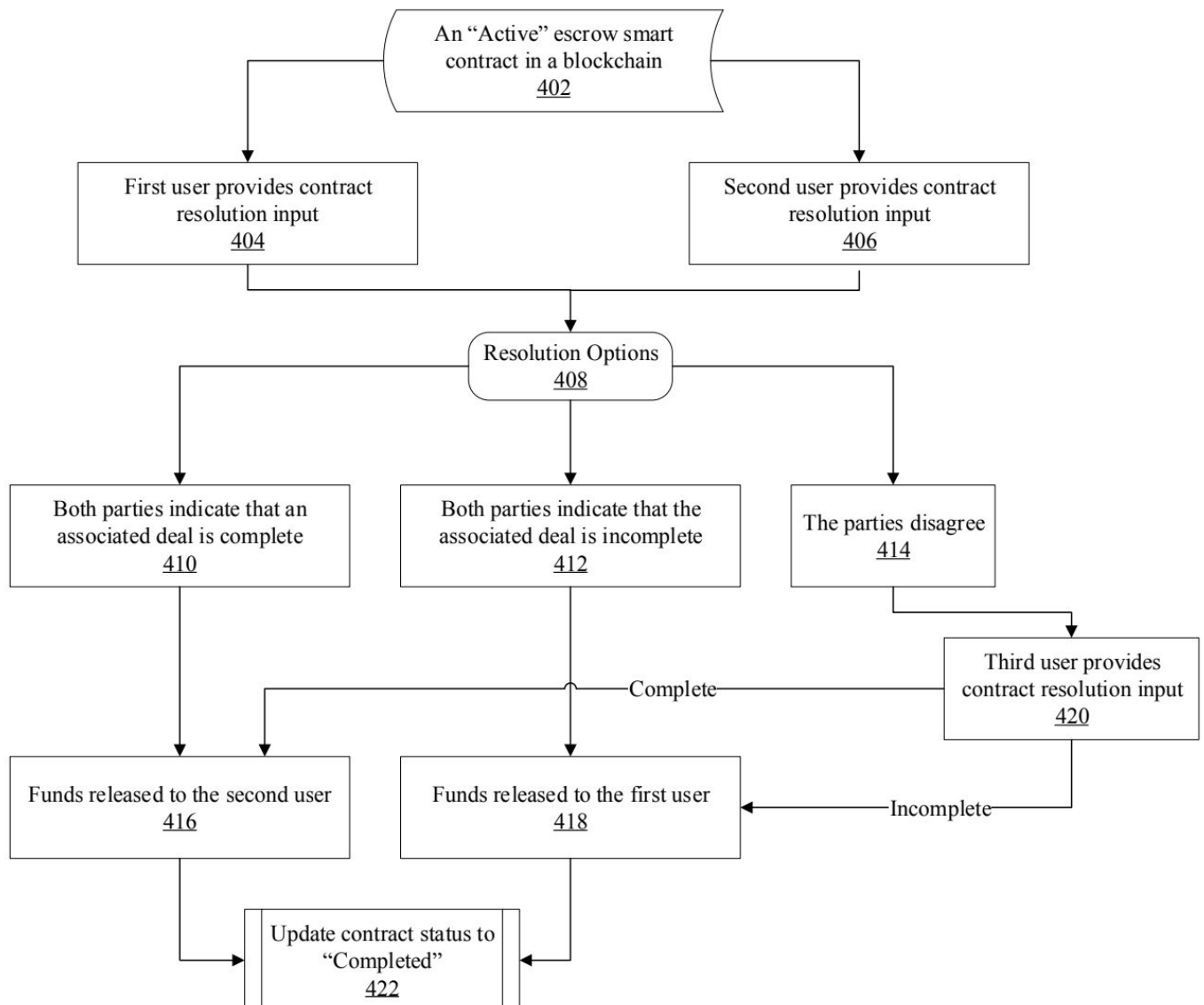
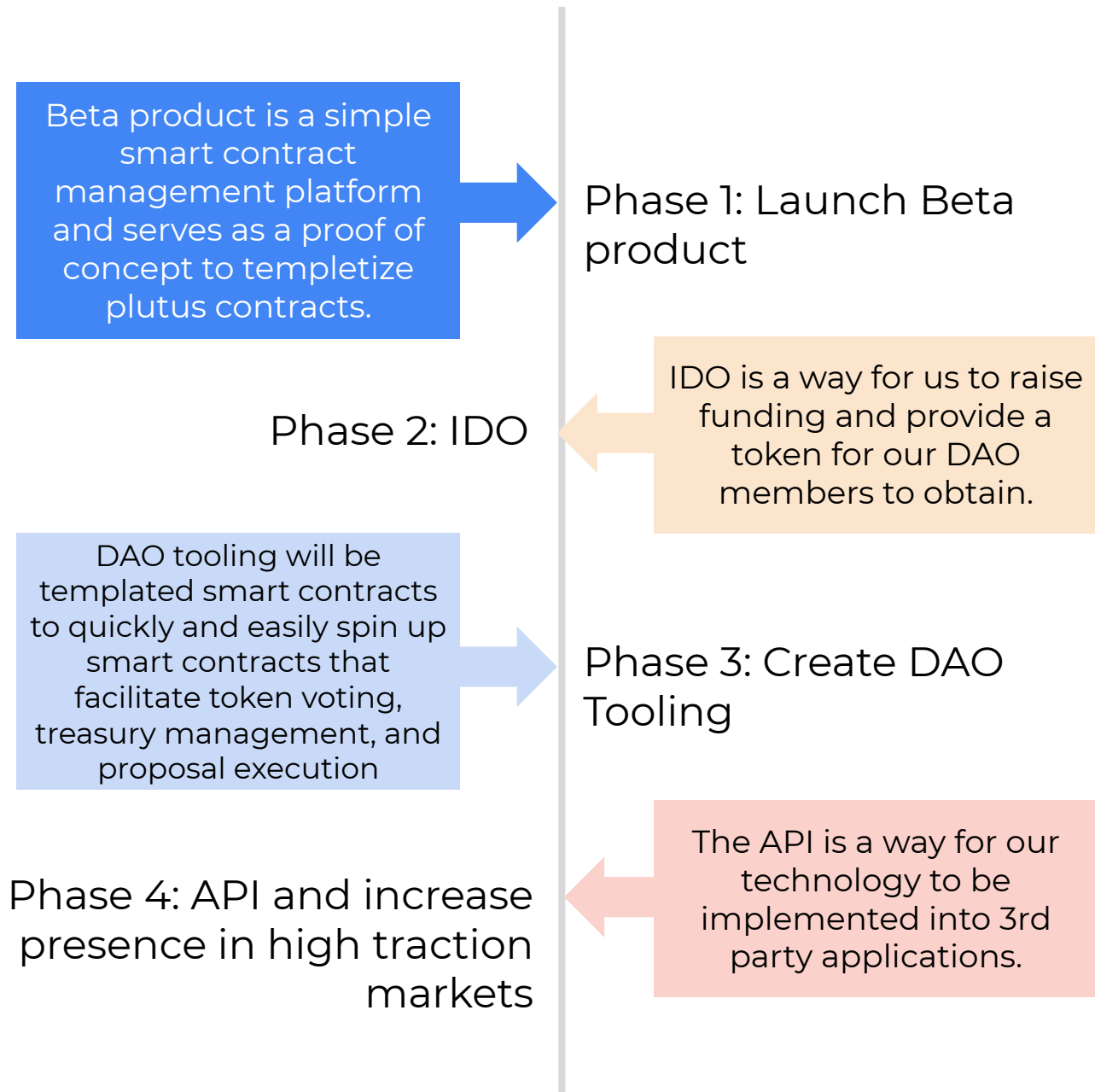
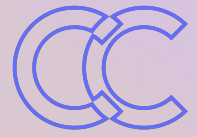
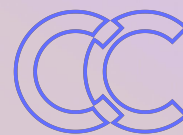


Fig. 2: Simple Escrow Smart Contract fund allocation logic

Roadmap



Tokenomics

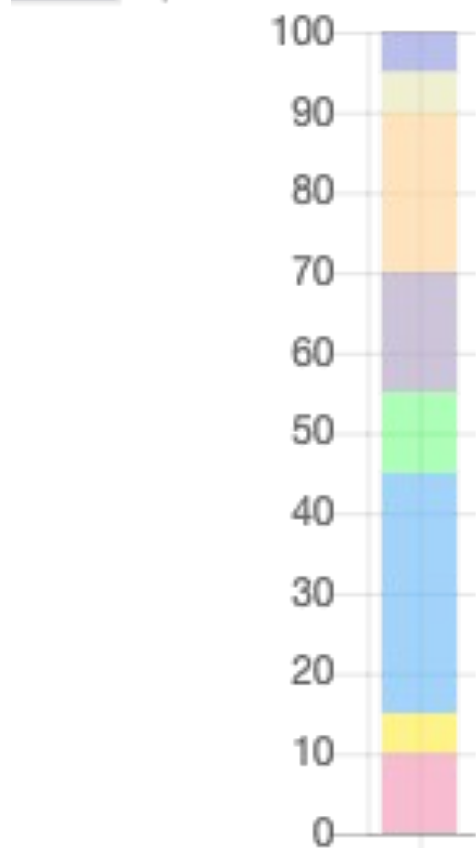


This is an overview of the token allocation for the Clear Contracts protocol. All funds contributed in the ICO will be used solely for the development, promotion, and growth of the Clear Contracts Platform.

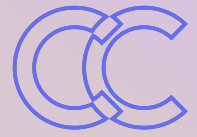
We will be updating this paper to clarify lockups, vesting, cliffs, and other relevant information.



- Seed Stage: 10%
- Private Sale: 5%
- Public Offering: 30%
- Liquidity Locked: 10%
- Ecosystem Fund: 15%
- Team: 20%
- Advisors: 5%
- Special Contribution, Incentives/Airdrops: 5%



Clear DAO Tokenomics



Token Utility

The Clear token serves two distinct purposes. *Governance and Staking.*

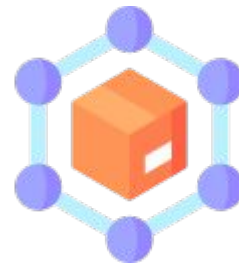
Governance

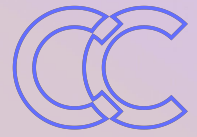
CLEAR token holders have complete control over the future direction of the protocol, including decision-making regarding development and product prioritization, proposal voting, and fee changes



Staking

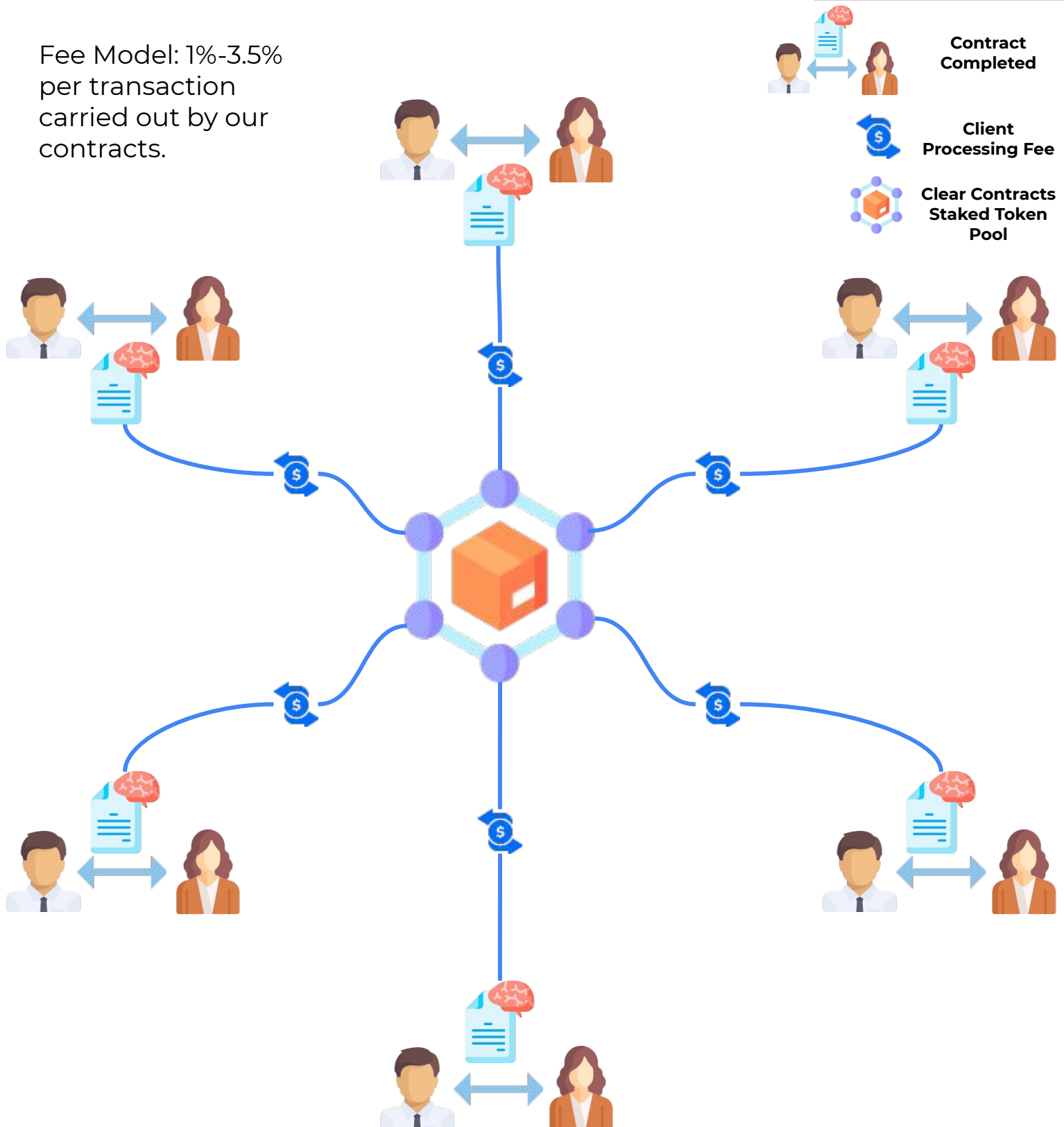
CLEAR token holders can stake their tokens to receive a portion of the transaction fees collected by the protocol

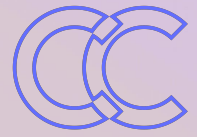




Clear Contracts Token Utility cont.

Fee Model: 1%-3.5% per transaction carried out by our contracts.

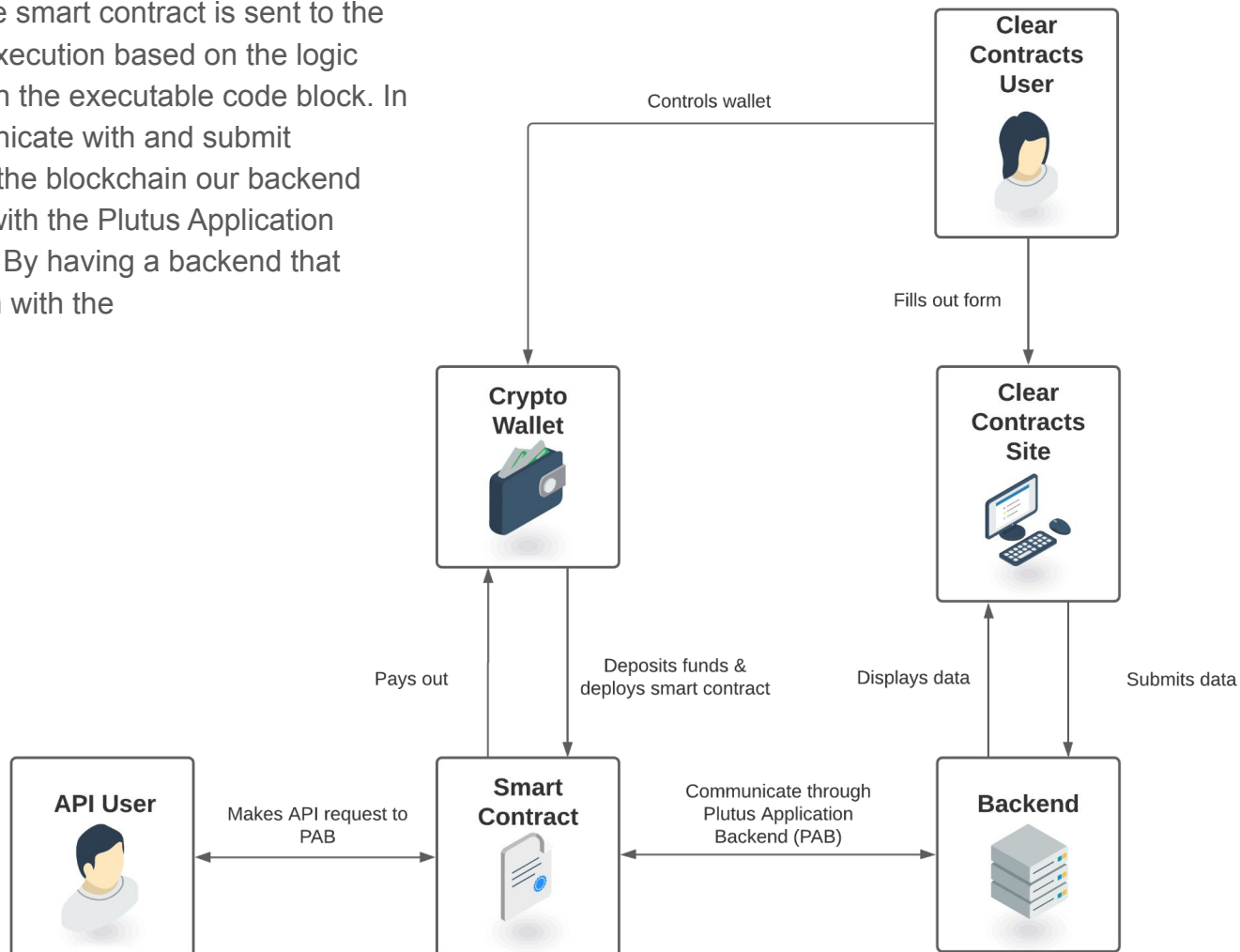


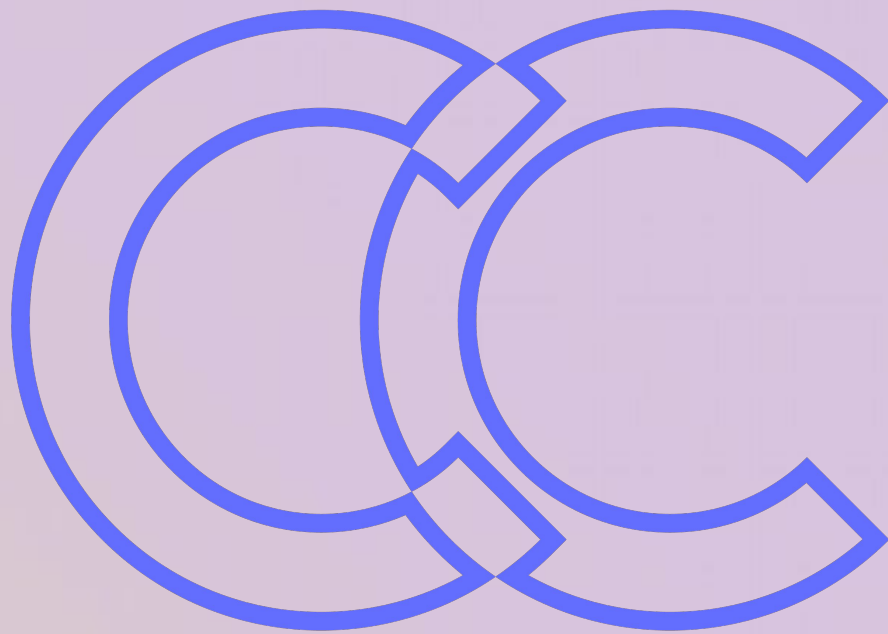


Platform Architecture

Our platform and services serve as direct access points to create, manage, and execute smart contracts on the Cardano Blockchain. In order to create a contract, a user must fill out a form online. The form inputs get collected and stored by our internal database. When all parties involved in the agreement accept their roles in the contract, our platform automatically populates the corresponding smart contract template and the smart contract is sent to the blockchain for execution based on the logic embedded within the executable code block. In order to communicate with and submit transactions on the blockchain our backend communicates with the Plutus Application Backend (PAB). By having a backend that works in tandem with the

PAB we are able to provide a unique solution that enables on chain actions to be triggered from our web application and services.





ClearContracts

Bringing smart contract technology to the
real world

