BIKECOIN TECHNICAL WHITE PAPER

Trustless Infrastructure for Smart Bikes

BIKECOIN
ECOSYSTEM

Trustless Infrastructure for Smart Bikes

Version 1.4

Eric Bui
Chief Blockchain Architect
EBui@VolataCycles.Com

Wee Keong Ng
Technical Advisor
awkng@ntu.edu.sg

MATTIA DE SANTIS
Chief Technology Officer
MDesantis@VolataCycles.Com

1. Introduction 3
2. BikeCoin Architecture Design 4
   2.1. BikeCoin Ethereum-based system 4
   2.2. Bike-Ownership Protocol employed the ERC721 token 5
   2.3. Automate payment process by Smart contracts and Payment Channels. 7
      2.3.1. The decentralized bicycle fleet service 8
      2.3.2. The decentralized peer-to-peer bike sharing network 13
   2.4. Data privacy solutions on BikeCoin Blockchain 15
      2.4.1. Anonymous Authority Layer empowered by Zero Knowledge Proof 15
      2.4.2. Linear Private Search Engine on Encrypted Data 19
   2.5. Discussion of future research 21
      2.5.1. Scalability solutions for public blockchain 21
      2.5.2. Other Security & Privacy For Data And Computation Solutions 22
3. IoT and Firmware Technologies In the Cycling Industry 23
   3.1. Emerging IoT trends in the cycling Industry 23
   3.2. Volata Technologies. 24
      3.2.1. The built-in computer on Bike 24
      3.2.2. Self-charging battery and e-bike. 27
4. Discussion 28

1. Introduction

Ethereum[1] is the decentralized computing platform where users deploy smart contracts to do computations; nobody controls Ethereum network and the users completely control their smart contracts – a trustless environment. It is a fantastic feature to build a decentralized application which user's smart contracts process transactions. Unlike the centralized application Facebook, decentralized applications various advantages in terms of data privacy, transparency and security.

Car makers BMW, General Motors, Ford and Renaults[2] launch blockchain research for automotive industry. In blockchain revolution, we, Volata Cycle – a "Tesla" in the cycling industry, believe that building a blockchain ecosystem for smart bicycles is needed than ever. We have developed BikeCoin – a trustless infrastructure with fantastic features such as asset tokenization, decentralization with completely data privacy. BikeCoin are empowered from these critical technologies:

- A permissioned blockchain - an off-chain layer to assist in processing transaction for Ethereum Smart Contract and along with a peer-to-peer storage layer, leveraging IPFS technology to store encrypted data.
- ERC721 to develop a Bike-Ownership protocol for identifying and verifying an owner of the bicycle.
- Our great Zero-Knowledge proof and cryptographic search algorithms on encrypted data to provide data privacy entirely on BikeCoin blockchain.
- BikeCoin token (ERC20) with state-channel technology to automate payment progress for a decentralized bike fleet management system and a decentralized peer-to-peer application for smart bikes which are the first decentralized application in BikeCoin ecosystem.

BikeCoin support cycling industry to make smart bikes safer, more affordable and more widely accessible to users.

[1] Ethereum https://www.ethereum.org/
[2] BMW, GM, Ford and Renault launch blockchain research https://techcrunch.com/2018/05/02/the-mobility-open-blockchain-initiative-bmw-gm-ford-renault/

2. BikeCoin Architecture Design

2.1. BikeCoin Ethereum-based system

Nowadays, many public blockchain projects are introduced with a higher transaction throughput than Ethereum. However, these projects only prove their technologies when their networks reach to Ethereum network size of 30, 000 miners. *Sharding* and *Plasma* are promising scalability solutions for Ethereum. These solutions, however, have not deployed on Ethereum main-net yet. Therefore, BikeCoin considers the scenario of a permissioned blockchain system – an off-chain layer to meet rigorous BikeCoin decentralized application requirements such as low latencies, immediate transaction finality, high performance and excellent scalability. In the permissioned blockchain, bicycle manufacturers are authorized and provide computing resource to process BikeCoin transactions.

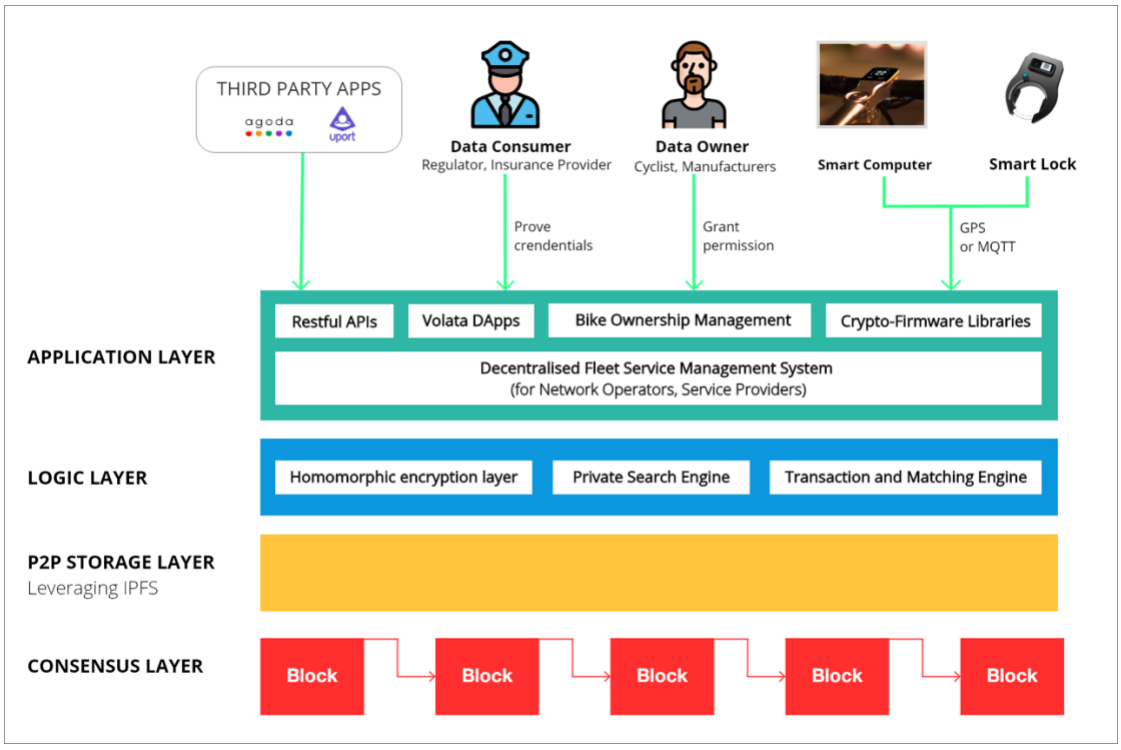

Figure 1 High-level System Architecture of BikeCoin

In the BikeCoin platform, all user's data should be encrypted at the client-side before being sent to the system. To maintain the access to the encrypted data, BikeCoin leverages IPFS peer-to-peer storage technology, a decentralized distributed hash-table (DHT) that is accessible through the blockchain, which stores references to the data but not the data themselves. DHTs have been widely used to coordinate and maintain metadata about

peer-to-peer systems. Kademlia[3] is a popular DHT that allows efficient lookup through massive networks, low coordination overhead, and resistance to various attacks by preferring long-lived nodes.

2.2. Bike-Ownership Protocol employed the ERC721 token

Ownership of smart bikes is rapidly increasing, but unfortunately also the theft of these expensive bikes. New smart computers or smart locks are on the market to remotely track and secure bikes. This provides opportunities for better theft prevention and new business models for bike rental. BikeCoin develop a Bike-Ownership protocol to share ownership information. Bike owners can prove their ownership and report the bike stolen through the blockchain. A smart computer or smart lock on the bike registers the position of the bike on the blockchain when they are locked. When a bike is stolen, the police is able to see the latest position to respond instantly. Insurance companies can automate their claim handling processing through a smart contract running on the blockchain

We have heard over and over again how ERC721's non-fungible, which tokens of the same class or contract can hold a different value. A value of one ERC721 Cryptokittie[4] does not equal the value of another ERC721 Cryptokittie because they are each unique. For this reason, we developed Bike-Ownership protocol[5] basing on ERC721 token from OpenZeppelin[6]; each ERC721 represents for each Bike, the value of the Bike-Ownership token is based on the metadata of token including bike technical information, picture, owner's information, manufacturer. The ownership is determined by an array of token *indexes* or *ids* that is mapped to Bike Owner address. The total supply of Bike-Ownership token is the length of array $allTokens$. The number of bicycles registered in the BikeCoin ecosystem is unlimited, as the maximum number of ERC721 Bike-Ownership token is $2^{256} - 1$.

```
25
26    // Array with all token ids, used for enumeration
27    uint256[] internal allTokens;
28
```

Figure 2 Array allToken stores all token ids

[3] Petar Maymounkove and David Mazieres http://www.scs.stanford.edu/~dm/home/papers/kpos.pdf
[4] Cryptokitties https://www.cryptokitties.co/
[5] Bike-Ownership protocol https://github.com/volatacycles/BikeOwnershipProtocol
[6] ERC721 standard OpenZeppelin https://github.com/OpenZeppelin/openzeppelin-solidity/blob/master/contracts/token/ERC721/ERC721Token.sol

```
109
110  /**
111   * @dev Internal function to add a token ID to the list of a given address
112   * @param _to address representing the new owner of the given token ID
113   * @param _tokenId uint256 ID of the token to be added to the tokens list of the given address
114   */
115  function addTokenTo(address _to, uint256 _tokenId) internal {
116    super.addTokenTo(_to, _tokenId);
117    uint256 length = ownedTokens[_to].length;
118    ownedTokens[_to].push(_tokenId);
119    ownedTokensIndex[_tokenId] = length;
120  }
121
```

Figure 3. Function addTokenTo in the smart contract BikeOwnershipprotocol.sol

```
248
249  /**
250   * @dev Internal function to add a token ID to the list of a given address
251   * @param _to address representing the new owner of the given token ID
252   * @param _tokenId uint256 ID of the token to be added to the tokens list of the given address
253   */
254  function addTokenTo(address _to, uint256 _tokenId) internal {
255    require(tokenOwner[_tokenId] == address(0));
256    tokenOwner[_tokenId] = _to;
257    ownedTokensCount[_to] = ownedTokensCount[_to].add(1);
258  }
259
```

Figure 4 Function addTokenTo in the smart contract ERC721BasicToken.sol

When bicycle manufacturers or bike owners register a new bike, the system will generate a random integer as a new token id and then call $addTokenTo$from the BikeOwnershipProtocol.sol to issue a new Bike-Ownership token.

```
31
32    // Optional mapping for token URIs
33    mapping(uint256 => string) internal tokenURIs;
34
```

```
58
59   /**
60    * @dev Returns an URI for a given token ID
61    * @dev Throws if the token ID does not exist. May return an empty string.
62    * @param _tokenId uint256 ID of the token to query
63    */
64   function tokenURI(uint256 _tokenId) public view returns (string) {
65     require(exists(_tokenId));
66     return tokenURIs[_tokenId];
67   }
68
```

```
 98
 99   /**
100    * @dev Internal function to set the token URI for a given token
101    * @dev Reverts if the token ID does not exist
102    * @param _tokenId uint256 ID of the token to set its URI
103    * @param _uri string URI to assign
104    */
105   function _setTokenURI(uint256 _tokenId, string _uri) internal {
106     require(exists(_tokenId));
107     tokenURIs[_tokenId] = _uri;
108   }
109
```

Figure 5 Each Bike-Ownership token mapped to a URI

Each Bike-Ownership token holds metadata which is URI of bike information such as manufacturer, bike frame information, smart computer lock information in the P2P storage layer.

Whenever users create a new Bike-Ownership token, the matching engine scan through all tokens by our searching engine (described in section 2.3); and if the matching engine finds any Bike-Ownership token with the same metadata, the system suggests users merge or transfer token between users. For example, Alice bought a Volata bicycle with smart computer IMEI: VOLATA123456789 from Bob and then Alice issue a new Bike-Ownership token for her bike. The matching engine detects that Alice's bike-ownership token has the same metadata (smart computer IMEI) with Volata manufacturer's bike-ownership token. The system notifies Alice and Volata for merge Bike-Ownership these tokens, e.g. burn the token 5 and transfer ownership of token 2 from Volata to Alice by a multi-signature contract.

2.3. Automate payment process by Smart contracts and Payment Channels.

µRaiden is a payment-channel framework for frequent, fast and free ERC20 token based micropayments between two parties. Whereas its big brother the Raiden Network[7] aims to allow for multi-hop transfers via a network of bi-directional payment channels. Raiden Networks allow us to transfer BikeCoin (BKC) near-instantly and low-fee by using digitally signed and hash-locked[8] transfers. In the payment channel, users have to setup on-chain deposit to open a payment channel, and then users can perform token transfer instantaneously, unlimited, as long as the net sum of their transfers does not exceed the deposited tokens; finally, users have to close the payment channel by calling functions $cooperativeClose(\ )\ or\ uncooperativeClose()$ in the smart contracts $TransferChannels.sol$ [9] which requires on-chain transactions. BikeCoin leverage Ethereum smart contracts and payment-channel technology to automate payment process among stakeholders.

[7] Raiden Network - https://raiden.network/
[8] Hashlock - https://en.bitcoin.it/wiki/Hashlock
[9] smart contract RaidenMicroTransferChannels.sol https://github.com/raiden-network/microraiden/blob/master/contracts/contracts/RaidenMicroTransferChannels.sol

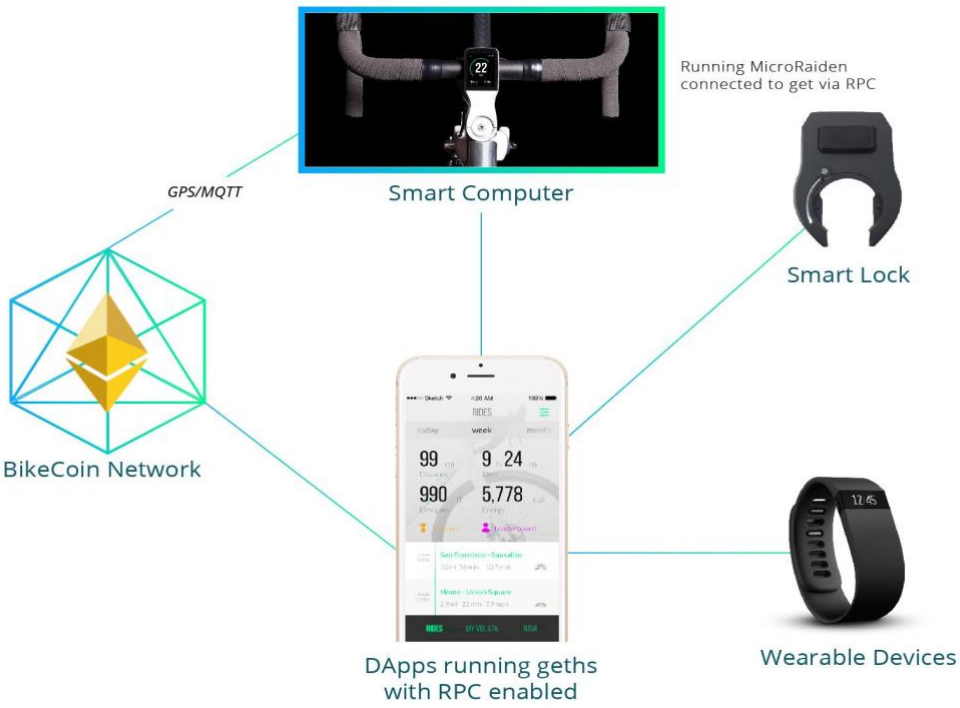

Figure 6 Networking topology with Raiden nodes in BikeCoin network

2.3.1. The decentralized bicycle fleet service

Bicycle fleet service is that Bicycle Suppliers (Volata Cycles) collaborate with Network Operators (bike retailers, bike hire companies or distributors) and Service Providers (hotels, premium resorts, co-working spaces, local councils, police forces, universities) to supply and maintain bike rental services to end-users. Bike fleet services business model have proved many advantages over Chinese dockless bike-sharing in terms of user experiences and friendly environment.

However, the current centralized fleet bike service systems expose many hassle-free and manual progress in bicycle maintenance service, auditing data and providing insurance service. To launch a bike fleet service together, bicycle suppliers, network operators, and service providers have to trust and store all data in the fleet management system which is centralized and lacks data privacy and security.

Moreover, a membership business model is not a fair way to share cost and revenue among bicycle suppliers, network operators and service providers in the current fleet services; for examples, service providers (hotels, premium resorts) do not have many tourists during un-holiday periods, and it is not necessary to replace a bike by the bicycle supplier if the bike has not reached to its limited number of use.

As a result, it limits bicycle suppliers, network operators and service providers in the world to collaborate with each other.

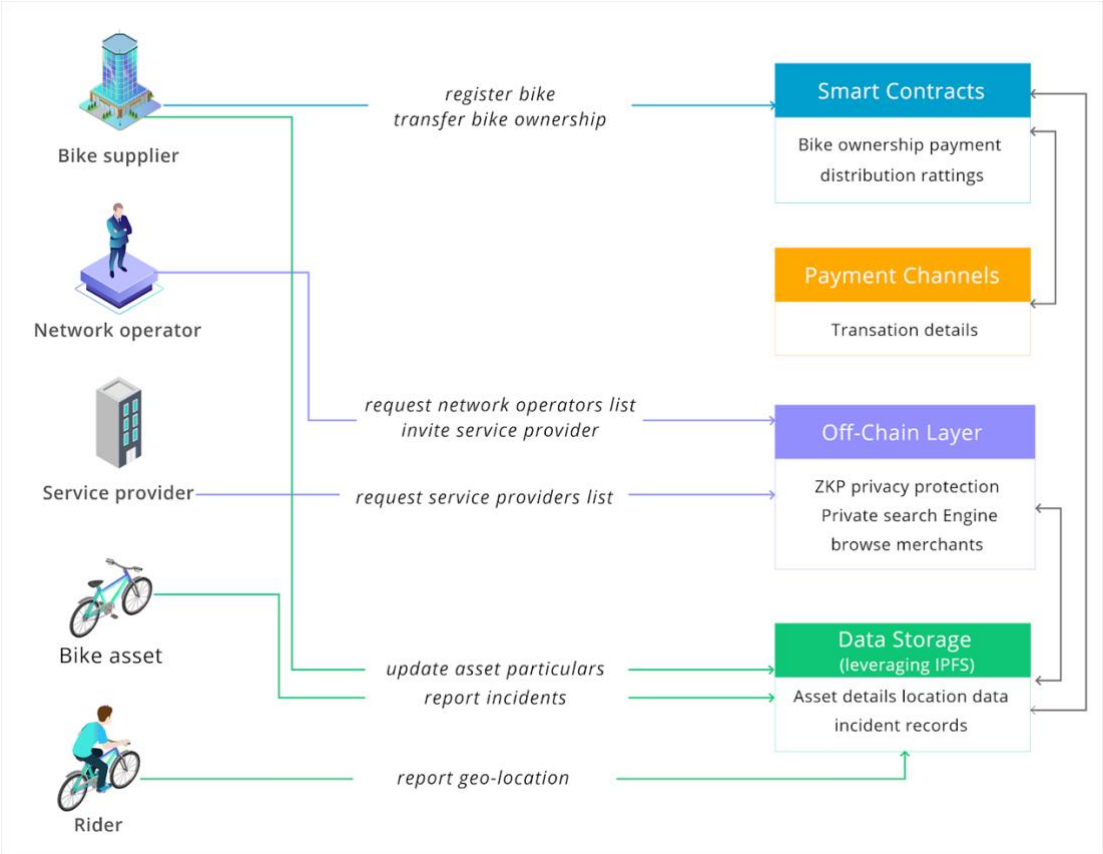

Figure 7 Asset Management & Partnering Features in Fleet Service

BikeCoin aims to build an open trustless blockchain ecosystem to decentralize bicycle fleet service where bicycle manufacturers, network operators, and service providers to collaborate each other with fair pay-per-use model smoothly.

Bicycle suppliers only need to update BikeCoin firmware library into the smart computer or smart locks of the bike. Their bikes directly interact with BikeCoin blockchain such as: closing a payment-channel, transfer BikeCoin (BKC) token as distributing revenue among stakeholders, recording location, bike status, user profiles or data into BikeCoin blockchain. In the future, BikeCoin will cooperate with IoT manufacturers to produce a built-in smart computer, smart locks working with BikeCoin ecosystem; as a result, it makes BikeCoin network widely adopted by bicycle manufacturers.

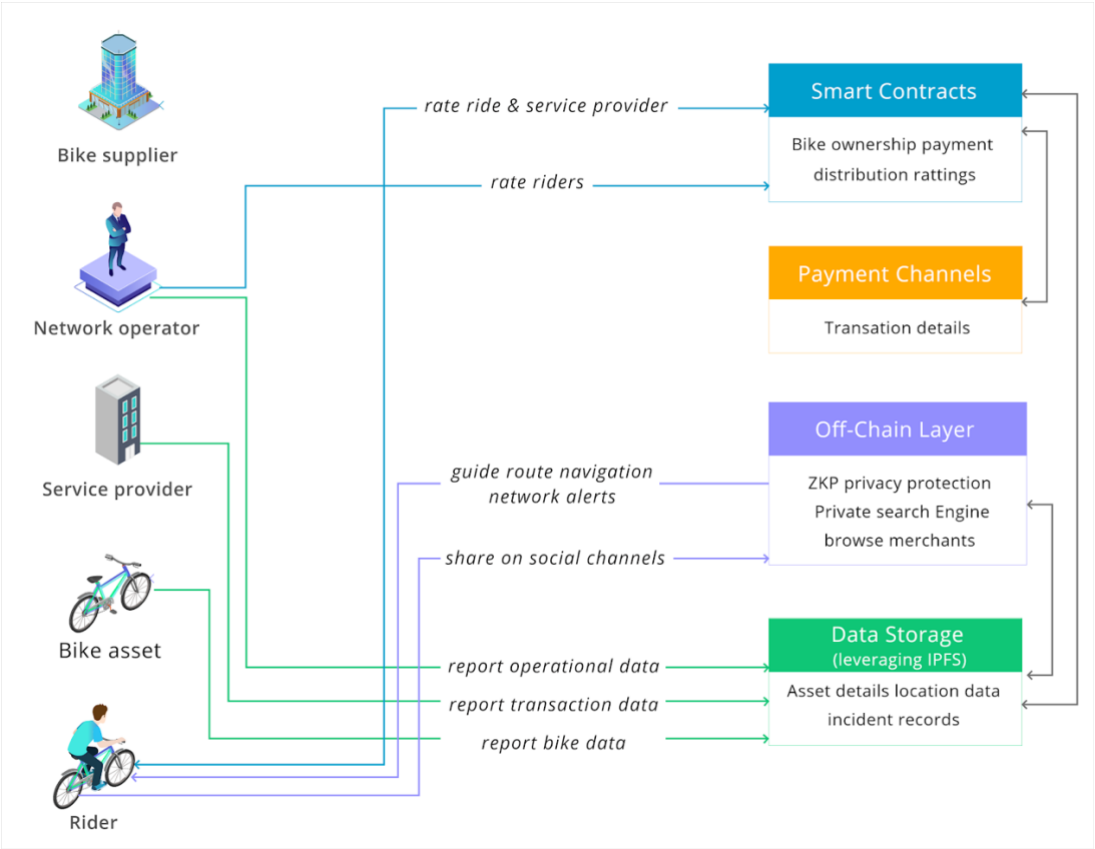

Figure 8 Data Sharing Features in Fleet Service

BikeCoin develop a decentralized fleet management system which is flexible and adaptable to several situations, from the e-bike rental business to public services as the first decentralized app on BikeCoin ecosystem. The decentralized fleet management system helps network operators to decide whether to redistribute the bikes to areas of higher demand or to identify special infrastructure needs in locations where most bikes are ridden. Moreover, network operators and service providers are able to do:

- Real-time monitoring of bicycle diagnostic
- Real-time bike geo-location
- Reporting statistics of usage and riding behaviour
- Scheduling maintenance services
- Restricted areas crossing alerts

Clients of service providers use a decentralized app such as Volata App to record performance, tracking data.

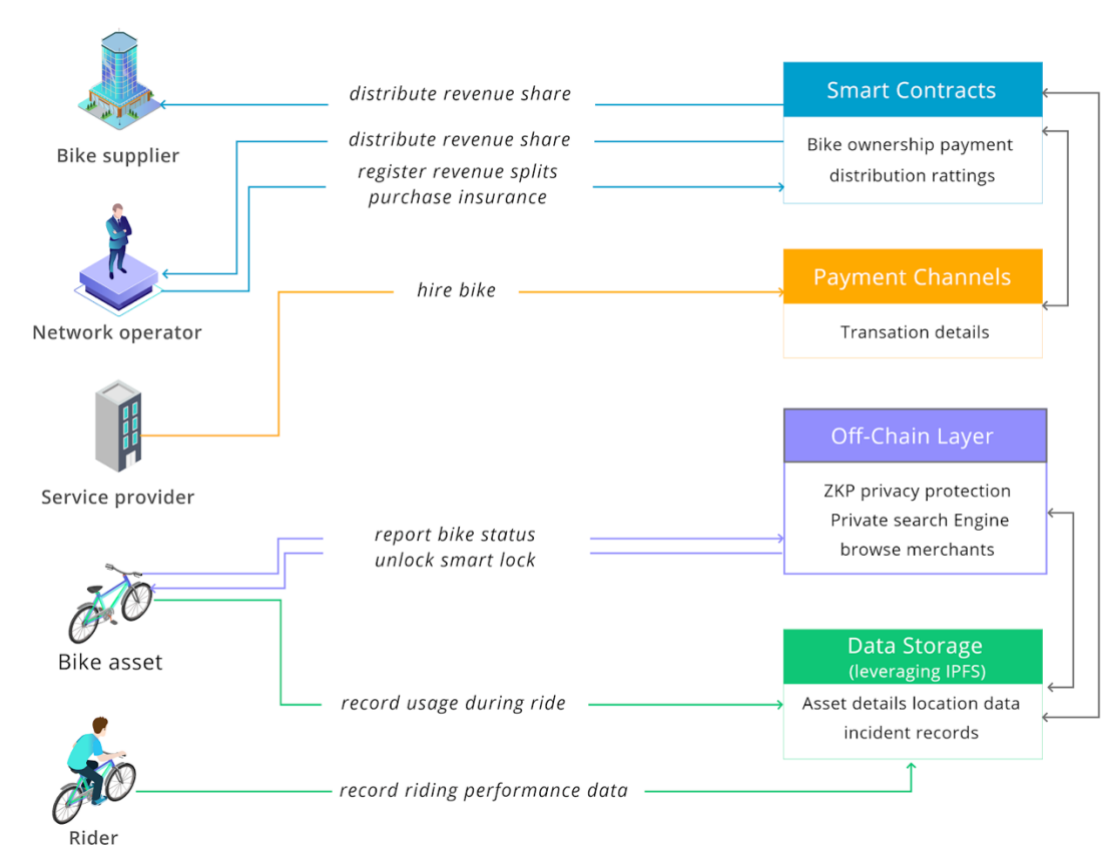

Figure 9 Transaction Support Features in Fleet Service

As an open platform, bicycle manufacturers, network operators, and service providers are able to reuse and customize easily the BikeCoin decentralized fleet management system easily, the mobile application to meet their business requirements. BikeCoin also incentive community developers to develop advanced data management tools like bicycle diagnosis tool, smart relocation methods…Insurance companies automate their fleet insurance payment progress conveniently by using BikeCoin smart contracts because all data are stored immutably on the BikeCoin blockchain.

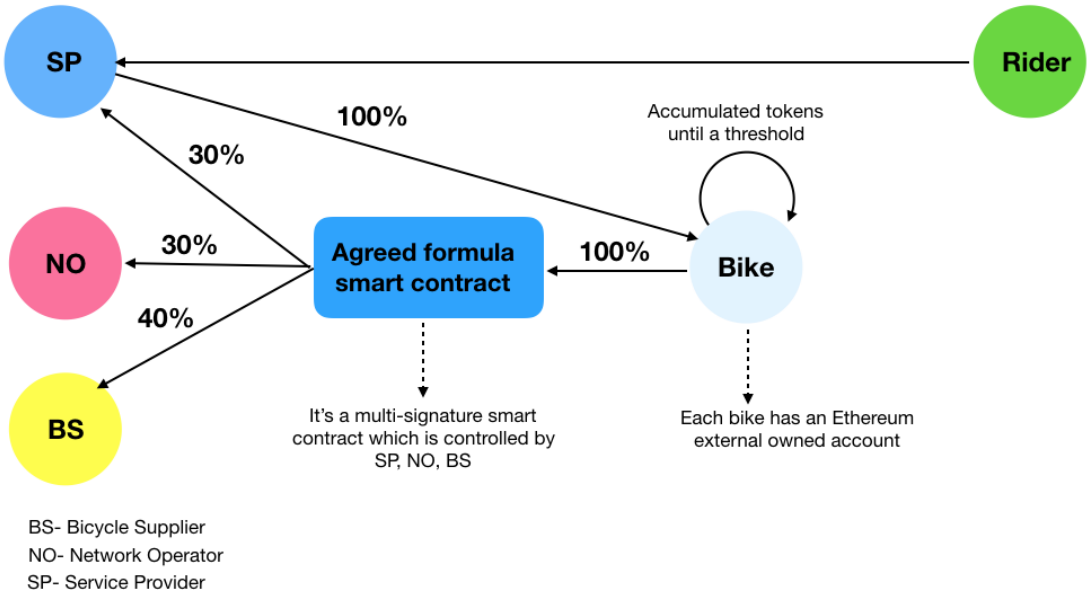

Figure 10 Revenue distributed by smart contracts.

We describe a specific scenario where bicycle manufacturer Volata Cycle Inc, a Singapore retailer Bob, and Marina Bay hotel to launch a fleet bicycle service.

1. Volata, retailer Bob, and Marina Bay have to settle an agreement regarding deposit, revenue split (supposed Marina Bay 30%, Bob 30%, and Volata 40%), maintenance terms, payment terms (such 8 cents per riding minute). All of these terms are recorded immutable by an Ethereum multi-signature contract in which any change has required an endorsement of three parties.
2. Marina Bay hotel has to buy BKC. Every morning, Marina Bay account automatically deposit 1000 BKC to open a Raiden payment channel with each Volata bike. It is an on-chain transaction; however, it is not required a real-time transaction.
3. Marina Bay guest use Volata app to rent, unlock and record and share riding data. After each riding, Marina Bay account automatically send a signed off-chain transaction directly to the Volata bike.
4. End of the day, Volata bikes, and Marina Bay automatically sign and close the payment channel to settle the payment on-chain. It is also not required as a real-time transaction.
5. Whenever Volata bike's balance account reaches the threshold, the Volata bike automatically distributes revenue among Volata Cycle Inc, Bob, and Marina Bay by transferring BKC to the agreed formula smart contract which created in step 1.

Therefore, In the BikeCoin ecosystem, bicycle suppliers, network operators, and service providers freely search and collaborate with each other without trust each other. All stakeholders follow BikeCoin smart contracts; is that service providers pay directly to the bike, and the bike only unlock if it got paid. The bike distributes revenue directly to all stakeholders by multi-signature smart contracts.

2.3.2. The decentralized peer-to-peer bike sharing network

As addressed in our business whitepaper, participating in a peer-to-peer bike sharing network, smart bikes solve not only urban transportation but also accelerate online bicycle purchase sale. Online purchasing users are able to experience riding a new bicycle before making online orders, so the price of smart bikes are more affordable by online direct sale from bicycle manufacturers to end-users.

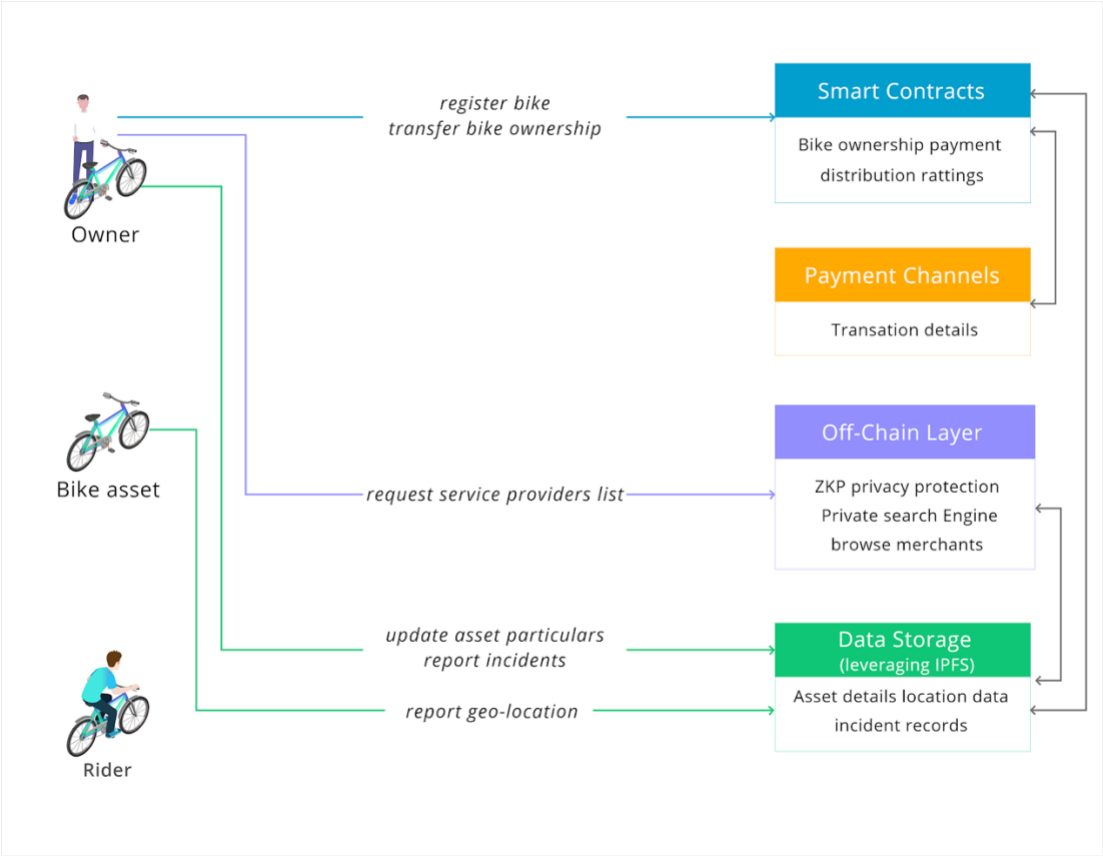

Figure 11 Asset Management & Partnering Features in the P2P service.

After registering their bike information by BikeCoin-Ownership protocol, bike owners are able to share various information with potential renters, eg bike information, image, reviews and history of riding. Bike owner list rental fee, terms, bike address and commission fee for the Dapps in Ethereum smart contracts. And each smart bike has an Ethereum externally

owned account[10] and must participate in a payment channel of Raiden network as a receiver.

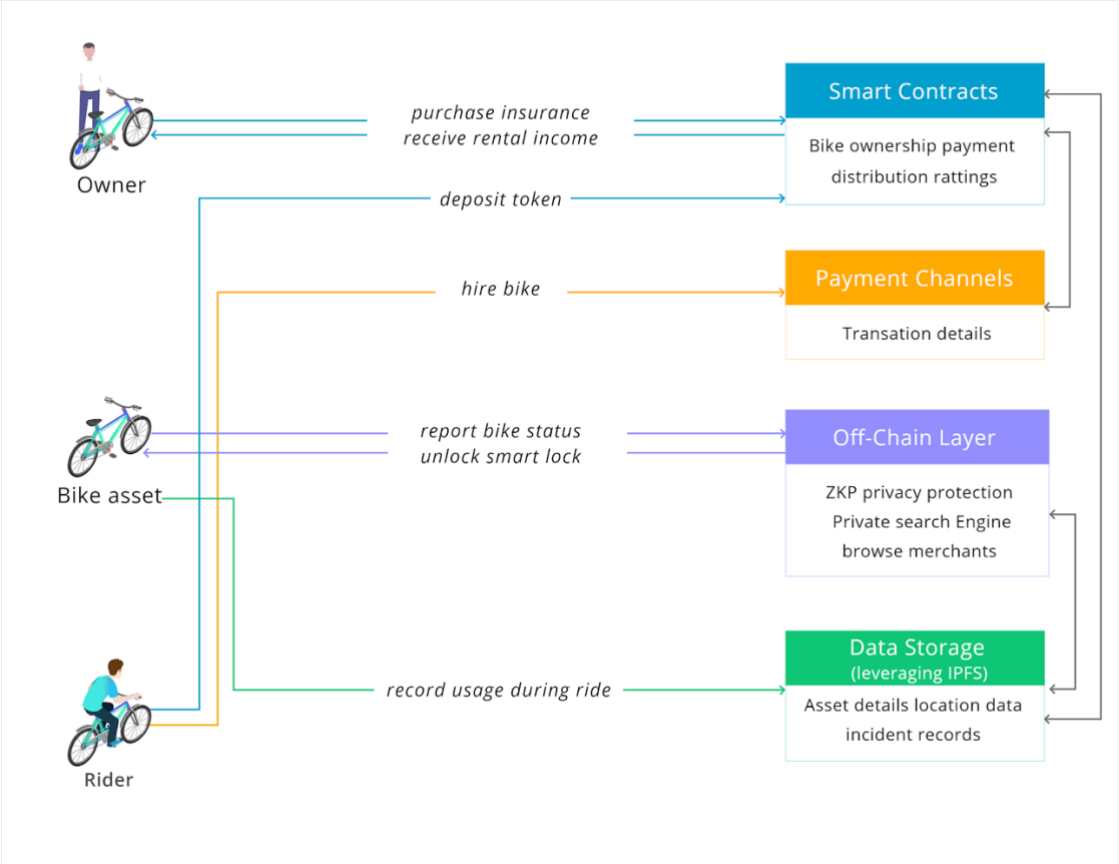

Figure 12 Transaction Support Features in the P2P service.

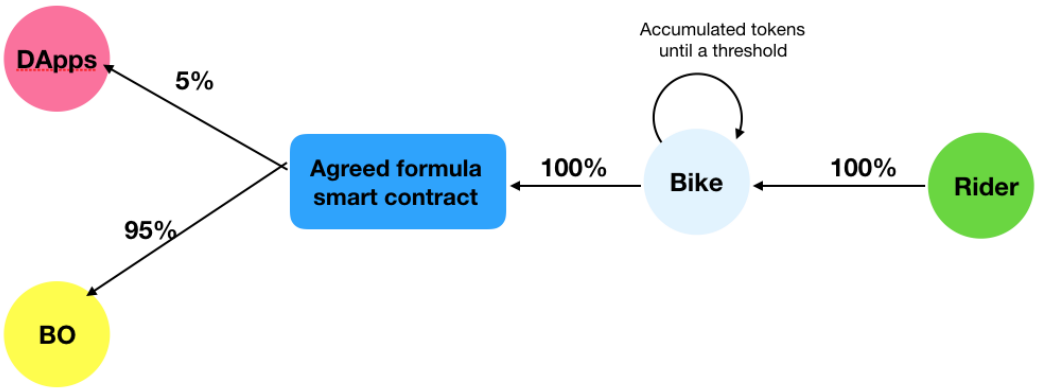

Figure 13 The payment flow in the P2P rental market

[10] Ethereum externally owned account http://ethdocs.org/en/latest/contracts-and-transactions/account-types-gas-and-transactions.html

Bicycle renters do registration on the decentralized sharing-economy applications which are built on BikeCoin blockchain. Bike renters have to deposit BKC and open a payment channel in Raiden network as a sender. After that, they are able to search bikes by our private search engine (discussed in section 2.3) and make a renting request. Each riding trip, bike renters pay directly to the bike account by sending a signed off-chain transaction on Raiden Network. Whenever the bike's balance account reaches the threshold, the bike automatically distributes revenue to bike owners and reward to BikeCoin DApps by transferring BKC to the agreed formula smart contract.

BikeCoin release Bike Ownership Management system for bike owners to manage all information, actions, orders and transactions; and BikeCoin sharing-economy DApps for bicycle renters to search bikes and control their data privacy settings. BikeCoin, however, encourages third-party developers, sharing-economy DApps to build on top BikeCoin protocols.

2.4. Data privacy solutions on BikeCoin Blockchain

Data on a blockchain is stored in a "distributed" ledger of records and is immutable- once recorded, data cannot be changed retroactively without the alteration of all subsequent blocks and a collusion of the network majority. It is an amazing feature of blockchain technology; however, all data is public. Although blockchain is an open anonymous system, we can trace the route of the coin and once it hit's some exchange; most of the time we know the actual person behind some bad things. We believe that bicycle manufacturers (Volata Cycle) and service providers (hotel, premium resorts or co-working space) do not want to their sale information, clients data are revealed to their competitors, and we also believe that rider, bike owners do not want to their performance data exposed to anyone.

Therefore, confidentiality for data is a fundamental requirement to adopt BikeCoin ecosystem. We design anonymous authority layers and cryptographic search engine modeled by smart contracts to manage encrypted data with complete data privacy.

2.4.1. Anonymous Authority Layer empowered by Zero Knowledge Proof

Anonymous Authority Layer allows a data consumer to obtain a credential from the data owner so that at some later points of time, he is able to construct a non-interactive proof for his credential. The blockchain nodes accept the request only if the attached proof is valid. The design of the

component is inspired by the idea of Zerocoin[11] with Zero Knowledge Proof technology.

The following example describes how the intuition of the anonymous authority layer. Let consider there is a public (i.e. everyone can access) physical bulletin board. To produce a new credential for the Insurance company A, the cyclist firstly generates a pseudonym S for the Insurance company A and commit S using a secure digital commitment scheme. The resulted commitment C can be opened by a random number r known by the Insurance company A. The cyclist pins C to the bulletin board, there is a set $S_C = (C_1, C_2, \ldots C_n)$of commitments in the board. At a later point, the Insurance company A is able to prove possession of such credential by producing two statements in zero-knowledge:

- He knows a commitment $C \in S_C = (C_1, C_2, \ldots, C_n)$.
- He knows the opening r for the commitment.

the BikeCoin is used as the public bulletin board. Both the data owner and the data consumer are able to access the public parts of the data stored on the blockchain. The public parts contain the commitments that we have described.

We now present a concrete construction using cryptographic accumulator proposed by Josh Benaloh[12] , and later improved by Jan Camenisch [13]. The accumulator scheme comprises four algorithms:

- $AccumSetup(\lambda) \rightarrow params$. Generates two primes p, q, computes $N = pq$, sample $u \in QR_N$. Output (N, u) as the parameters.
- $Accumulate(C) \rightarrow A$.On a set of primes $C = \{c_1, \ldots, c_n\}$, outputs accumulator $A = u^{c_1 \ldots c_n} \bmod N$.
- $GenWitness(v, C) \rightarrow \omega$.Input a prime number $v \in C$, outputs a witness$\omega = Accumulate(C - v)$.
- $AccVerify(\omega, v, A) \rightarrow \{0,1\}$.Verifies $A = \omega^v \bmod N$The security of the scheme based on the harness of Strong RSA and Discrete Logarithm assumptions.

The description of the anonymous authority layer consists of four algorithms:

1. $Setup(1^\lambda) \rightarrow params$. On the input parameter λ, run the algorithm $AccumSetup(1^\lambda)$to obtain (N, u). Generate primes p, qsuch that $p = 2^w q + 1$for $\omega \geq 1$.Let G be the subgroup of $Z_q^*$and select two random generator g, h such that $G = [g] = [h]$.

[11] Ian Miers, Christina Garman, Matthew Green, Aviel D. Rubin http://zerocoin.org/media/pdf/ZerocoinOakland.pdf
[12] Josh Benaloh, Michael de Mare https://www.microsoft.com/en-us/research/wp-content/uploads/1993/01/owa.pdf
[13] Jan Camenisch, Anna Lysyanskaya http://cs.brown.edu/~anna/papers/camlys02.pdf

2. $GenCred(S, params) \rightarrow (c, skc)$. Given pseudonym $S \in Z_q^*$, select a random $r \in Z_q$ and compute $c \leftarrow g^S h^r$such that c prime and $c \in [A, B]$, where $2 < A$ and $B < A^2$. Set $skc = r$ and output (c, skc), submit $c$to the blockchain.
3. $ShowCred(params, S, c, skc, Sc) \rightarrow \pi_S$ Given data consumer pseudonym S, a credential c and its secret key skc, compute $A = Accumulate(params, S_c)$ and $\omega = GenWitness(params, c, S_c)$ and output the following proof of knowledge:

$$\Pi_s = ZKSoK\{(c, w, r, S) : AccVerify((N, u), A, c, \omega) = 1 \wedge c = g^S h^r$$

4. $VerifyCred(params, \pi, S_c)$. Given a proof Π_S, and the public set of credential S_C, first compute $A \rightarrow Accumulate\,(params, S_C)$, then verify that Π_S is the aforementioned proof of knowledge on c, S_c. if the proof verifies successfully, output 1, otherwise output 0.

The zero-knowledge proof appears in step 3 of the scheme is a non-interactive proof that only requires one round of communication. Camenisch present an interactive zero-knowledge proof of knowledge that an accumulator contains a committed value. The construction of the non-interactive proof in step 3 leverages Fiat-Shamir transform on the interactive proof.

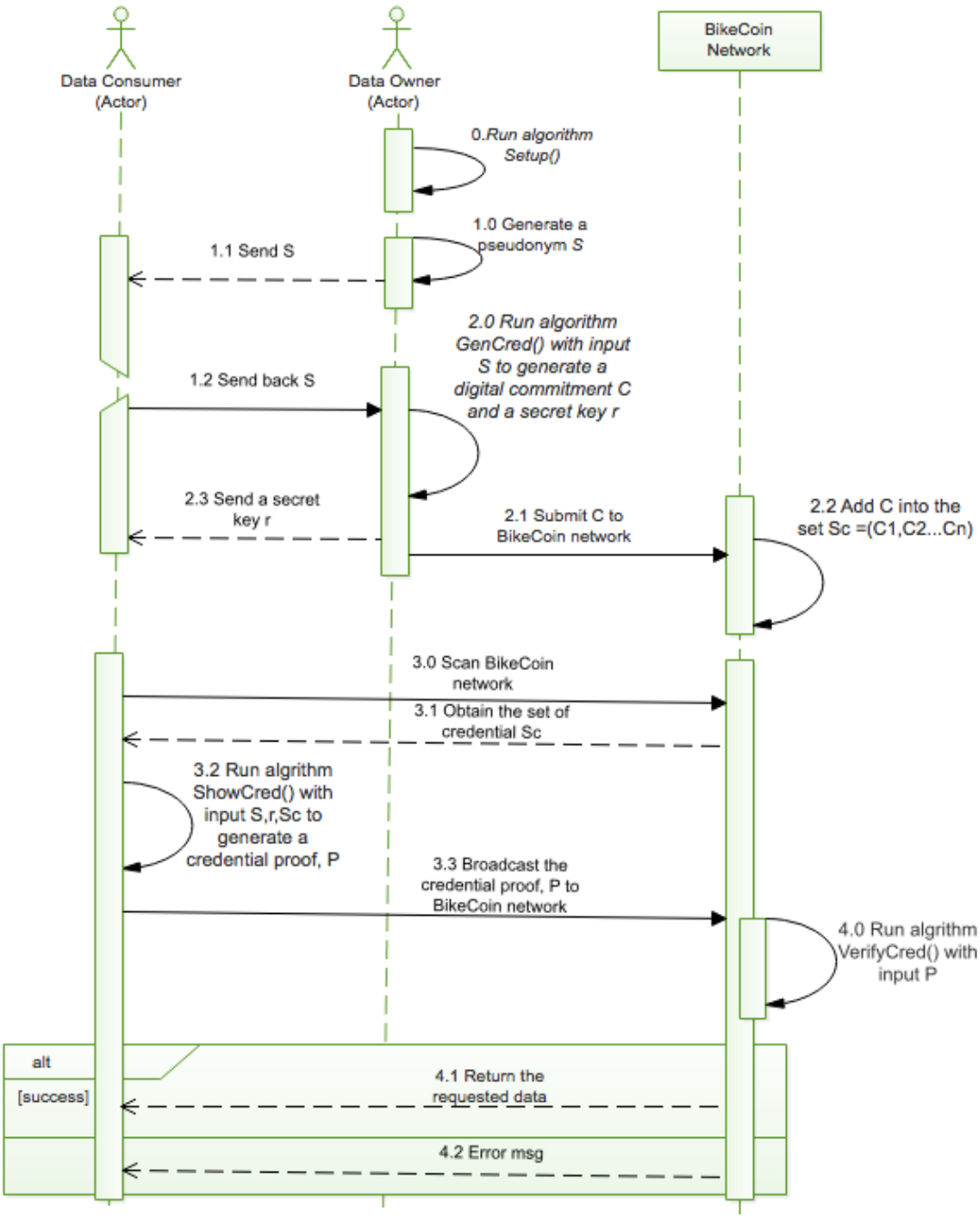

Figure 14 Sharing Data in the Anonymous Authority Layer

The $Setup$ algorithm is performed by the data owner to generate system parameters. Next, when data consumer wishes to obtain a credential for data access, he sends a request to data owner together with his pseudonym S. At this point, data owner runs $GenCred$ routine on this input S to generate a digital commitment and its secret key skc.

When the data consumer wishes to show his credential, he first scans through the blockchain (i.e. the bulletin board) to obtain the set S_c consisting of all credential issued by the data owner. He then runs the $ShowCred$routine

to generate a credential proof, and broadcast it to the blockchain nodes for verification. The blockchain nodes also collect the set of credentials in the blockchain and validate the proof using the $VerifyCred$ algorithm. The credential certification is accepted if the last routine outputs 1.

The data consumer and the blockchain nodes are both required to compute $A = Accumulate(params, S_c)$ that requires linear scan the blockchain data. The complexity of the protocol linear to the size of the number of registered data consumers.

Now with our robust anonymous authority layer, data owner is able to freely share data with data consumer without exposing her/his identity address. Unlikely Facebook, no information about users is collected and all your data is encrypted and accessed by smart contracts which users control. Riders and bike owners (data owners) freely upload riding performance data, heart rate, calories or identity documents and share to their friends, on social apps Twitter, Facebook…(data consumers) without exposed to anyone else. Bicycle manufacturers, hotels, premium resorts are confident to collaborate and exchange data with each other on BikeCoin network without revealed by their competitors.

2.4.2. Linear Private Search Engine on Encrypted Data

The crucial requirement of a search engine is fast without exposing private data. Our solution is that the meta-data (i.e the fingerprint) of the encrypted data are stored in the blockchain, and only the authorized clients are able to search via it. The authorization process is done by the Anonymous Authority Layer presented in the section 2.4.1

Users data stored, shared and can be accessed through a DHT in a peer-to-peer storage as discussed in section 2.1. The access key is computed using a hash function (i.e. the fingerprint of the data). The blockchain does not store the actual data content, however, it maintains the access key data so that the data consumers are able to link to the DHT using blockchain.

We denote EKS as the encryption scheme that supports keyword search. The data owner appends to the access key a list of EKS ciphertext of each keyword and stores it in the blockchain layer. A data D with keywords $W_1, W_2, \ldots, W_n$ is stored in the blockchain layer under the structure: $H(D) \,||EKS(W_1)\,||\ldots||EKS(W_n)$. An authorized data consumer is able to produce a certain trapdoor Γ_ω that enables a smart contract to test on each data entry whether one of the keywords associated with the access key (e.g. the document) is equal to the word W. Given a trapdoor and EKSciphertext, the blockchain nodes can only test whether $W = W'$, and nothing else.

A typical keywords search cryptosystem consists of four general algorithms:

1. $KeyGen$:generates cryptosystem key.

2. $Trapdoor$: produces trapdoor T_W for a keyword W
3. $Encrypt$:produces a EKSciphertext for keyword W
4. $Test$: tests whether keyword in the trapdoor is matched to the EKS ciphertext.

In BikeCoin ecosystem, an additional algorithm is required for the data owner to produce a secret search key for the data consumer. We denote that algorithm $KeyDerive$. Three algorithms $KeyGen, Encrypt$and $KeyDerive$ are performed by the data owner, while $Trapdoor$ is run by the data consumer to generate trapdoor, and finally, the $Test$ algorithms is done by the smart contracts or the blockchain peers. We modify the protocol proposed by Raluca Ada Popa[14] to adapt our problem settings.

We start the protocol description by reviewing a few concepts related to bilinear maps. We will use the following notation: $G_1$and $G_2$are two (multiplicative) cyclic groups of prime order $p, g_1$is a generator of G_1 and $g_2$is a generator of G_2. A bilinear map is a map $e: G_1 \times G_2 \rightarrow G_T$with the two following properties: $(1)\ Bilinear$: $\forall u \in G_1, v \in G_2$ and $a, b \in Z$,then $e(u^a, v^b) = e(u, v)^{ab}$, and $(2)\ Non-degenerate$: $e(g_1, g_2) \neq 1$.

We denote $H: \{0,1\}^* \rightarrow G_1$ and $H_2: G_T \times G_T \rightarrow \{0,1\}^*$ be two random oracles, and g_1, g_2, g_T are respectively the generators of groups G_1, G_2, G_T. The private keyword search system consists of five algorithms as the follows:

1. $KeyGen: k \leftarrow Z_p$.
2. $KeyDerive(k, s): k_s \leftarrow g^{\frac{k}{s}}$.
3. $Trapdoor(w, s): T_w \leftarrow e(H(w)^s, k_s)$.
4. $Encrypted(k, w): Random\ r \leftarrow G_T.\ Output: c = (r, H_2(r, e(H(w), g_w)^k)$.
5. $Test: Parse\ c = (r, h)$. Test whether $H_2(r, tk) = h$.

The data owner generates a secret key k for keyword encryption EKS, and derives data consumers' search keys. Each data consumer poses a secret $s \in Z_p$, which can be generated by a mapping from his pseudonym with the data owner. Using $k$and s, the data owner computes a search key k_s for the data consumer so that later he can use it for trapdoor construction.

The correctness of the protocol follows the two equations: $tk = e(H(w)^s, g_2^{\frac{k}{s}}) = e(H(w), g_2)^k$, and $H_2(r, tk) = H_2(r, e(H(w), g_2)^k)$
The above scheme has data hiding and token hiding properties. Data hiding (privacy) requires that the semi-honest adversary is not able to distinguish between ciphertexts of two values not matched by some token. Token hiding (privacy) requires that the adversary cannot learn the keyword that one searches for. **The complexity of the protocol is linear to the number of data set stored in the BikeCoin ecosystem.**

[14] Raluca Ada Popa and Nickolai Zeldovich https://people.csail.mit.edu/nickolai/papers/popa-multikey-eprint.pdf

2.5. Discussion of future research

2.5.1. Scalability solutions for public blockchain

As our technical viewpoint, a blockchain based system is as secure and robust as its consensus model. The best consensus model is to have the best balance of three following factors to have the best balance of three following factors:

- Decentralization: Any node freely participates in processing transactions, publishing a block without the use of central authority or service.
- Security: The system has to prevent double-spends, keep data in sync. There are no conflicts when data get merged. All nodes see same data at the same time.
- Scalability: The system has to sufficient transaction throughput to serve planet-scale or enterprise-scale needs, especially when the network size increases.

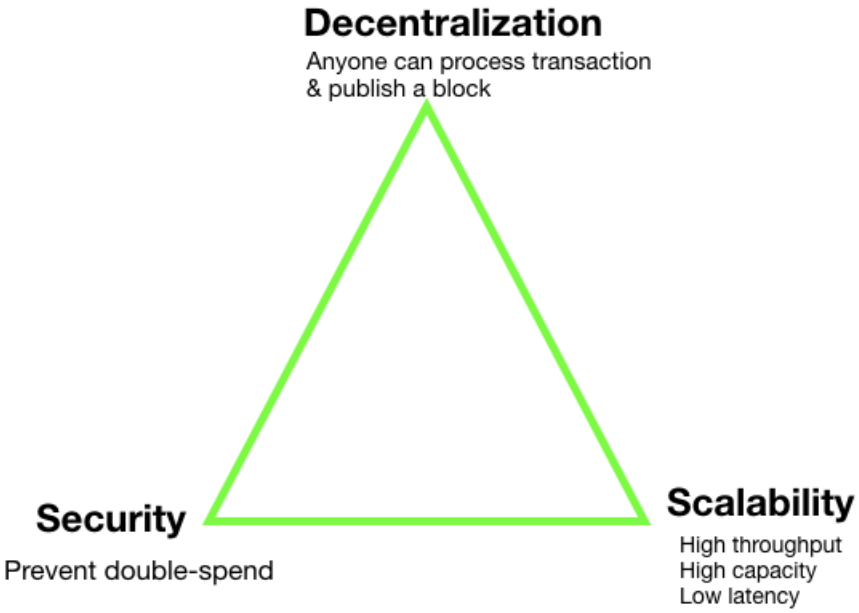

Figure 15. Three crucial factors of blockchain consensus algorithms

	PoW	PoS	PoET	BFT and variants	Federated BFT
Blockchain Type	Permissionless	Both	Both	Permissioned	Permissionless
Blockchain	BitCoin, Ethereum	Omisego, BitShare, PeerCoin	IntelLedger	Hyperledger Fabric	Ripple & Stellar
Transaction finality	Probabilistic	Probabilistic	Probabilistic	Immediate	Immediate
Transaction rate	Low	High	Medium	High	High
Scalability of peer network	High	High	High	Low	High
Trust model	Untrusted	Untrusted	Untrusted	Semi-trusted	Semi-trusted
Adversary Tolerance	<= 25%	Depends on specific algorithm used	Unknown	<= 33%	<=33%

Figure 16. A comparison of blockchain consensus mechanisms

Sharding[15] and Plasma[16] of Ethereum blockchain are promising solutions to achieve the best balance of three factors. BikeCoin actively embrace these promising scalability solutions.

2.5.2. Other Security & Privacy For Data And Computation Solutions

We showed how to do storage and random queries by zero-knowledge proofs and cryptographic algorithms. However, there are other security and privacy for data and computations solutions: Secure Multi-party Computation and Trusted Execution Environment (TEE). Secure multi-party computation enables multiple parties to jointly compute a function over inputs without disclosing said input and Enigma[17] team focus on this solution. Trusted Execution Environment solutions are mostly based on Intel SGX [18] and are intended to use for Intel SawtoothLake[19] blockchain. As our technical expertise, three solutions Zero Knowledge Proof, Secure Multi-Party Computation and Trusted Execution Environment have own advantages, and none of them compete with each other entirely. Therefore, BikeCoin actively collaborates, and research with other blockchain startups to develop these solutions in BikeCoin ecosystem.

[15] Sharding in Zilliqa blockchain https://docs.zilliqa.com/whitepaper.pdf
[16] Plasma https://plasma.io/
[17] Enigma https://enigma.co/
[18] Intel SGX https://en.wikipedia.org/wiki/Software_Guard_Extensions
[19] Intel SawtootLake Architecture Overview https://sawtooth.hyperledger.org/docs/core/releases/0.7/sawtooth_developers_guide/architecture_overview.html

3. IoT and Firmware Technologies In the Cycling Industry

3.1. Emerging IoT trends in the cycling Industry

The bicycle is the most energy-efficient means of human transport ever invented, and the future is all about sustainability. It solves all the problems of urban mobility and sedentary life. With rapidly increasing a number of active users in the bike-sharing market, bicycle manufacturers and tech giants actively apply innovative IoT technologies to smart bikes with better user-experience or safety.

Ofo and Huawei have worked together to build the NB-IoT-based[20] smart shared bike lock solutions providing lower power consumption, better coverage, and lower latency; or Mobike, AT&T, and Qualcomm[21] collaborate on Mobile IoT smart bike sharing technology. Moreover, IoT-Smart bike technologies are well-known recognised by tech enthusiast community such as with the Remote Controller and Lock system, the MAT received the HONOREE prize at CES 2018 Innovation Awards in the *Vehicle Intelligence and Self-Driving Technology category*. [22]

Figure 17 the Remote Controller and Lock system

[20] Smart Shared Bicycle Lock http://www.huawei.com/minisite/iot/en/smart-bike-sharing.html
[21] Mobike, AT&T and Qualcomm collaborate https://www.prnewswire.com/news-releases/mobike-att-and-qualcomm-collaborate-on-mobile-iot-smart-bike-share-technology-300516548.html
[22] MAT awarded at CES 2018 innovation awards http://www.esb.bike/mat-awarded-ces-2018-innovation-awards/

3.2. Volata Technologies.

Cars have been revolutionized by software, to improve both safety and user experience.

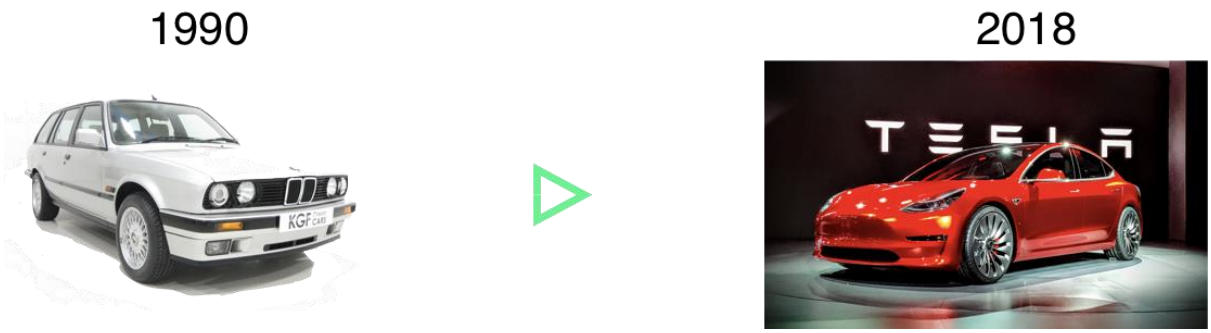

Figure 18 Car Revolution

Volata is the first bicycle company to design hardware, software, and take care of distribution and service. Comparatively, we want Volata to do within the cycling industry - what Tesla has done in the automotive industry.

3.2.1. The built-in computer on Bike

The built-in computer is easily controlled via thumb joystick and shows smartphone notifications (such as calls, text messages) on its. Volata provides the user with all those features that he or she needs when using the bike for commuting or recreational riding, with a focus on safety, and providing them significant rewards while riding.

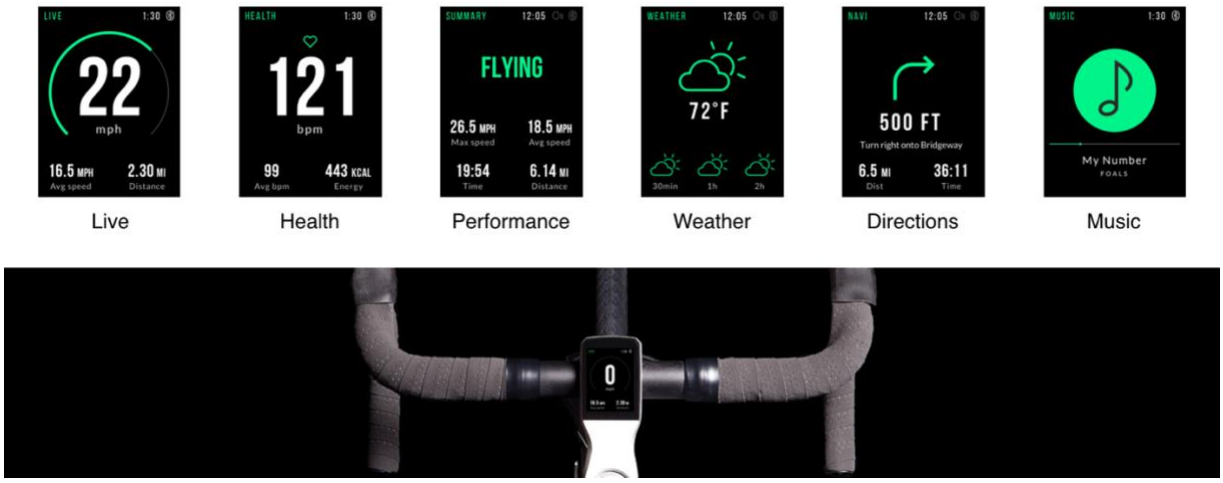

Figure 19 Built-in applications on Volata Smart Computer

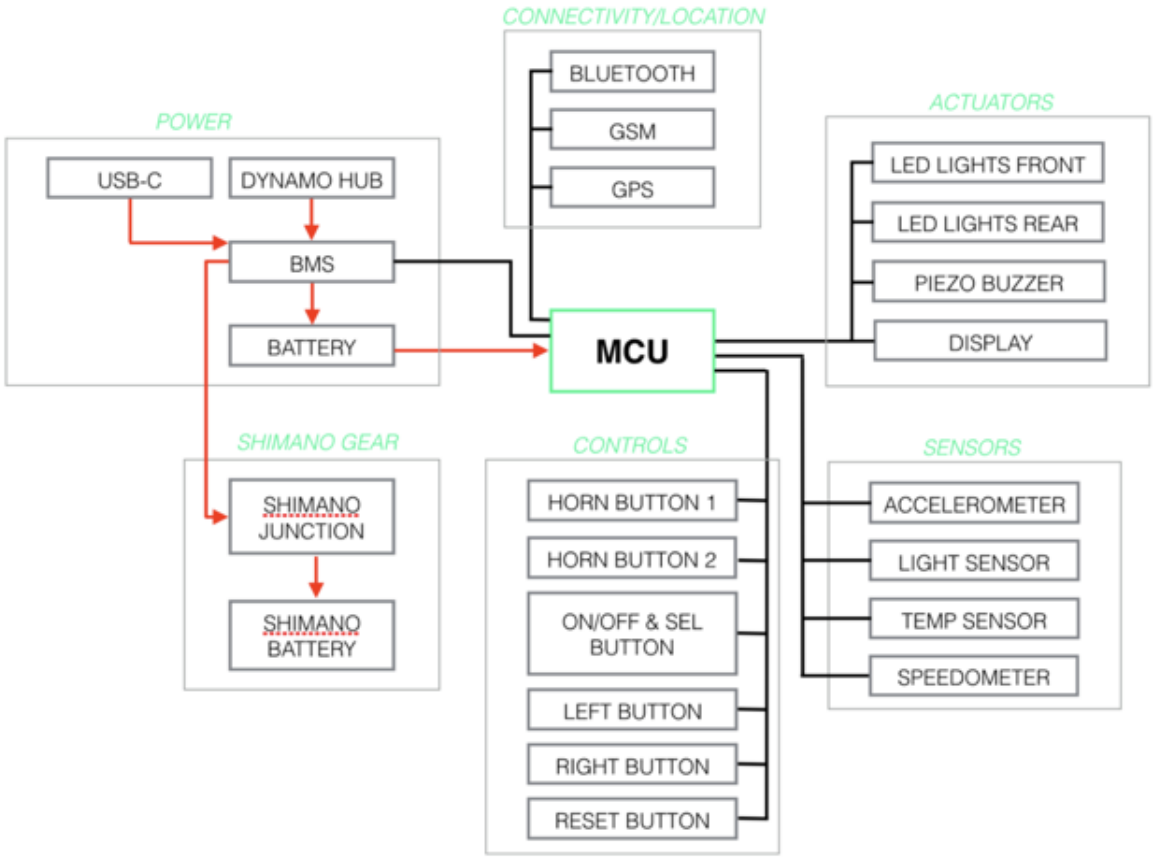

Figure 20 Overall Architecture design

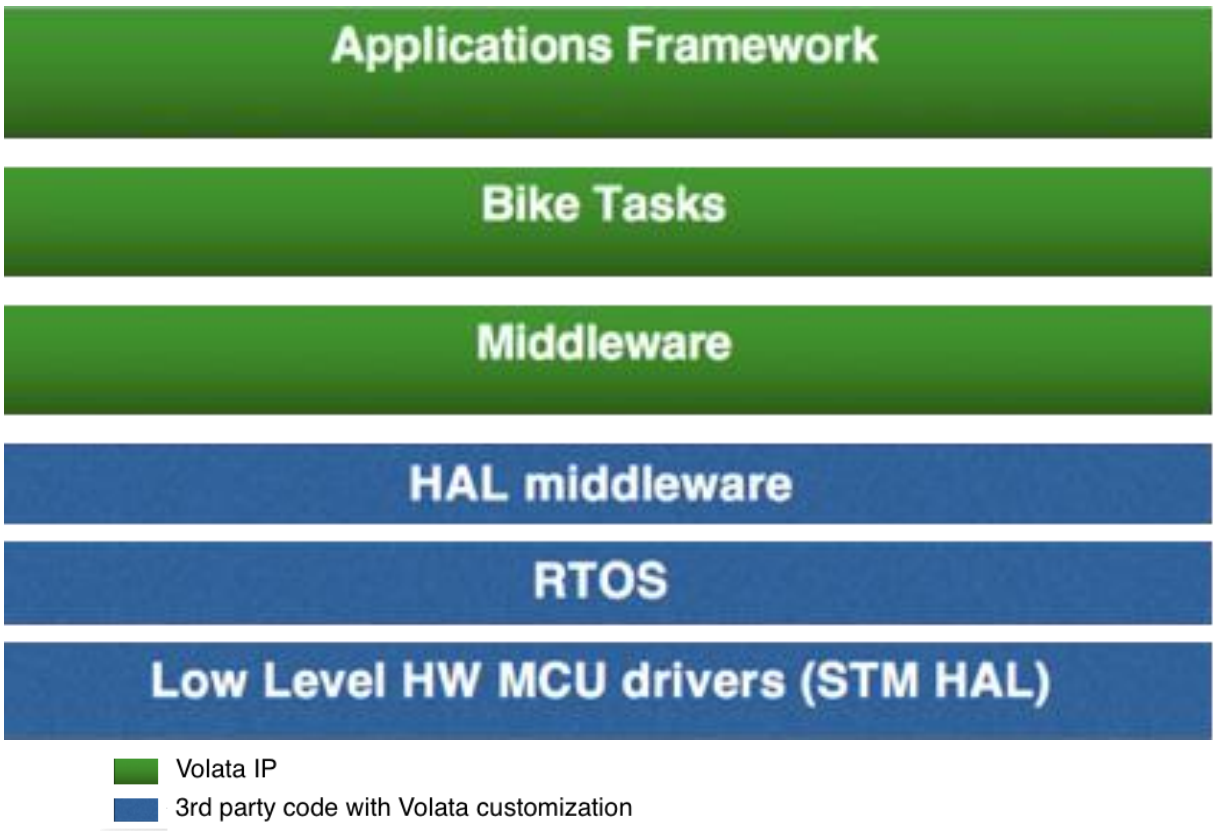

Figure 21 Software Components

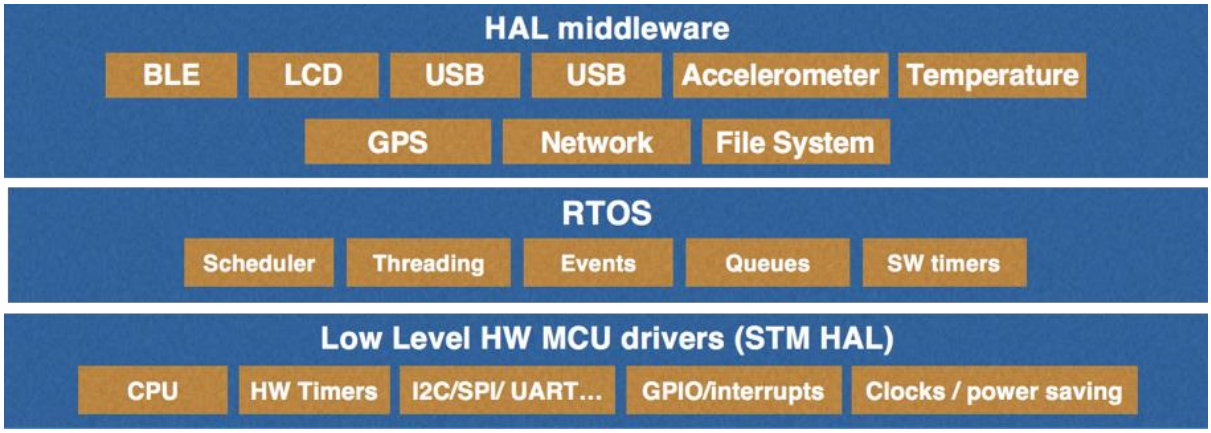

Figure 22 RTOS and Drivers

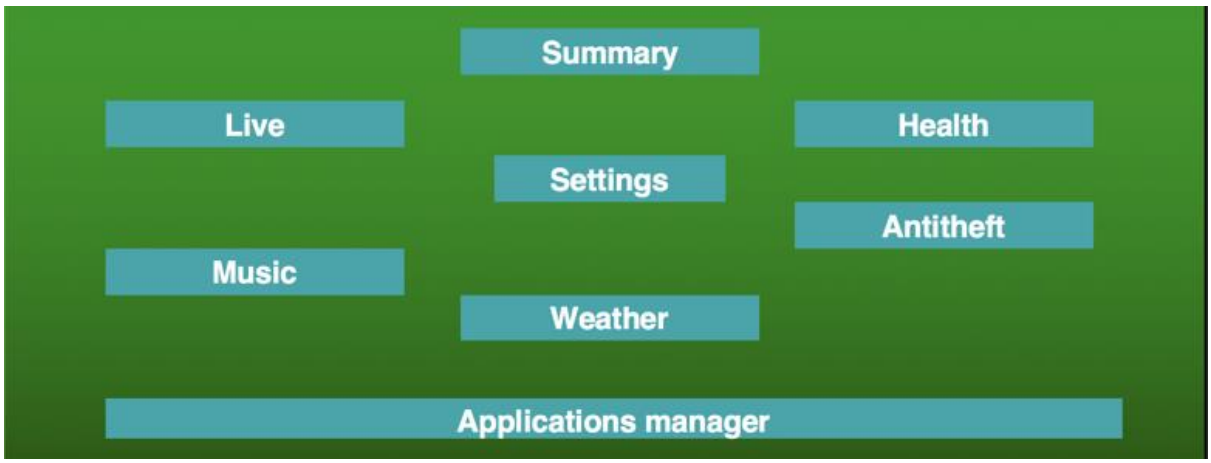

Figure 23 Application Framework

We build a BikeCoin application in Volata Applications framework, and each bike has an Ethereum externally owned accounts. Thus, Volata bikes can receive BKC, open/close payment channels and automatically distribute BKC among stakeholders as revenue split.

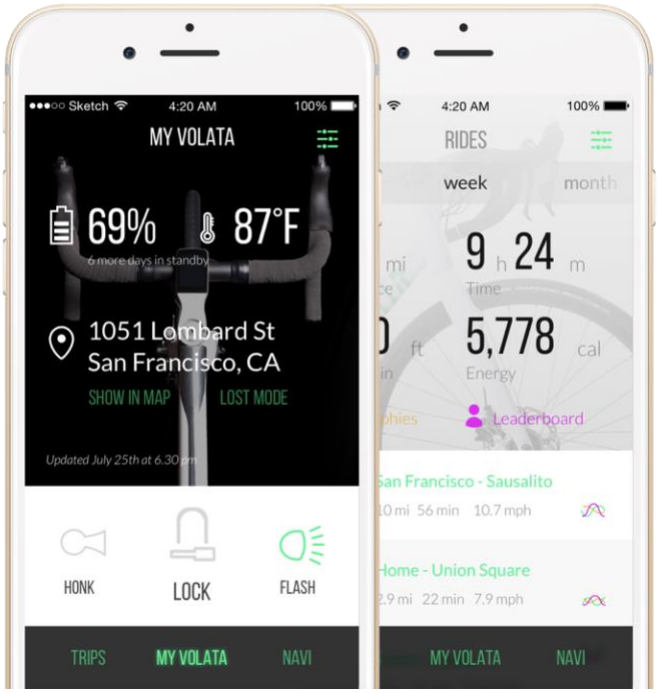

Figure 24 Volata App

The bicycle connects to the internet (via GSM) and to the rider's smartphone - through Volata's app, the rider can: riding data summary, personal trophies, a timeline of all trips, set destination for directions and share their performance with the community. Like Teslas and BMWs, Volata bike can be located, monitored, and remotely locked or unlocked at any time from a smartphone and let the user receive a notification if somebody tries to steal the bike, allowing to GPS-track it lives by built-in GPS and GPRS

3.2.2. Self-charging battery and e-bike.

We've extended our integration-core also on Volata's hardware components, using a belt instead of a chain, and an internally geared-hub instead of the traditional derailleurs. This translates into a huge reduction of maintenance, which is fundamental, especially on the commuting side.

Figure 25 Thanks to a front dynamo hub, the battery never needs to be charged

Volata Cycles also designs a number of bicycle extensions (carrier rack, frame bag, etc.) to adapt the bike to different user needs. Electric bicycles will play a crucial role in the transition to sustainable

transportation. As a next step, Volata will also expand to make electric bicycles with the same level of integration as our current premium model.

4. Discussion

In the sharp contrast to cloud computing technology (a centralized system), blockchain enables us to build a system without a trusted third-party system which all stakeholders equally collaborate with each other in the frictionless way- decentralize everything[23]. Blockchain technology, however, is still at the early stage and there are some challenges such as scalability and data confidentiality, need to be solved to meet rigorous requirements of enterprise applications.

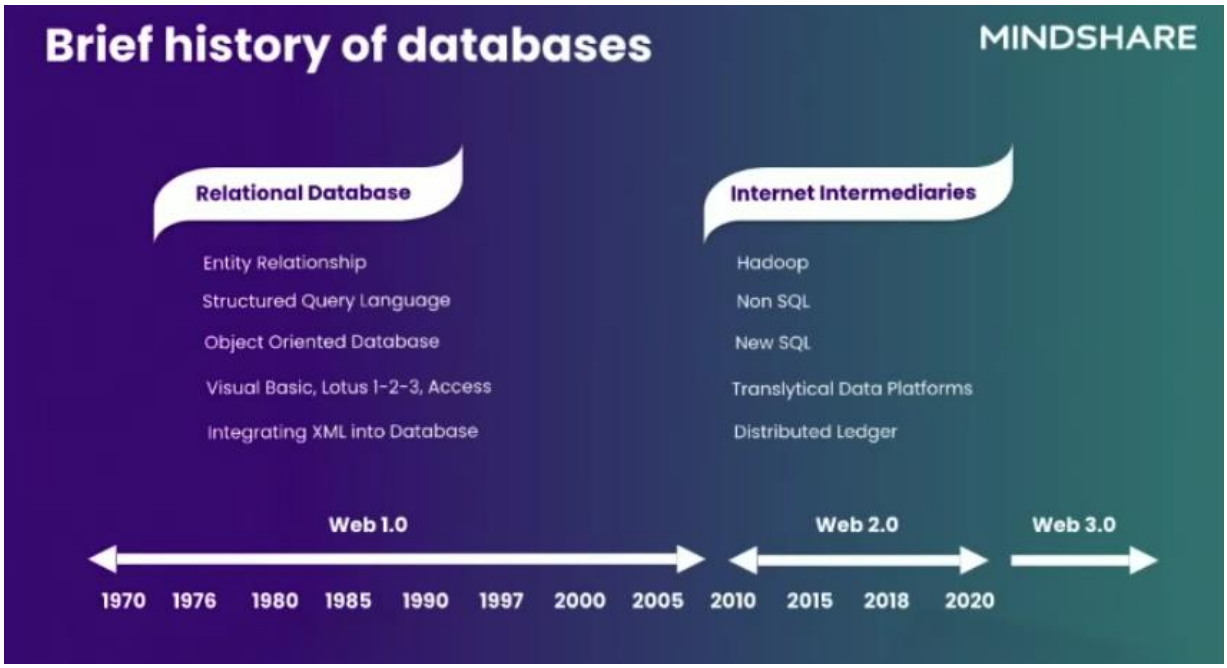

Figure 26 Brief history of databases by MINDSHARE in Zilliqa talk [24]

Sharding (in Zilliqa[25]) and Plasma are these most promising scalability solutions to achieve the best balance of three factors (decentralization, security, and scalability); but it takes time for these solutions ready to be used e.g. deploying on the main-net. Therefore, BikeCoin uses the permissioned blockchain as an off-chain layer to assist Ethereum on-chain transactions. The on-chain smart contracts handle bike registration, the link of data point and remuneration while the off-chain smart contracts do routing, data storage, and search transactions. This direction helps

[23] Decentralized Everthing – Vitalik Talks https://cyber.capital/2017/10/11/decentralizing-everything-with-ethereums-vitalik-buterin-disrupt-sf-2017/
[24] Zilliqa talk: Sharding in Action & Security by Design https://www.youtube.com/watch?v=_7Rv1Q5exbE
[25] Zilliqa blockchain - https://www.zilliqa.com/

BikeCoin deliver a fast and realistic technical roadmap, and the project is less risky and dependent on the public blockchain technologies like Ethereum or Zilliqa.

Security and privacy for data and computations are one of the hottest challenging research topics in blockchain technology. There are four main approaches for this challenges: Zero-knowledge proof (using cryptography and local computation), Fully homomorphic encryption (using pure cryptography), secure multi-party computation (sMPC using cryptography and distributed machine) and trusted execution environment (using secure hardware – Intel SGX). Each approach has its advantage and disadvantages in term of performance, functionality, integrity, privacy, and interoperability; none of them compete with each other ultimately. BikeCoin actively collaborates with world-class blockchain research teams to develop other privacy approaches (e.g. sMPC or secure hardware) for the whole BikeCoin ecosystem.