Contents

Introduction

Blockchain is about decentralization, but this also means difficult usage for end users. Some people have issues using computers and smartphones, how are they supposed to work with private keys, wallet browser addons and smart contract function calling? The answer is, they simply won't use it because it is „*too complicated*" and they „*don't have time*" to learn how to use it. Still, blockchain projects are aiming for a full decentralization, and wondering why almost no real users are using their solutions. We think decentralization is great. But it's not working on every part of a project/solution. In our eyes, the *perfect solution* is a mixture of both, a **centralized part** and a **decentralized part**. The data itself is safe on the blockchain, but this doesn't help if only a few have enough knowledge to access it. There has to be a *front-end* solution that is easy to use, **centralized**. We think this is the only possibility to increase the adoption of cryptocurrencies. **Tokengateway.io** and its future projects aim to build a bridge between the normal web and the Blockchain with the final goal to increase the adoption of cryptocurrencies for businesses and end users.

Executive summary

Tokengateway.io offers a variety of *API functions* that make it easier to work with *ERC20 tokens*. No matter if the goal is to use an own *ERC20 token* as a payment method in an online shop, create an exchange, provide a wallet function to users, send tokens to millions of airdrop participants or building a bridge between a website currency and an *ERC20 token*. **Tokengateway.io** has you covered with all needed functions. There are functions to send any *ERC20 token*, send *Ethereum*, rent and control *Ethereum addresses*, receive any *ERC20 token* via webhook/Instant Payment Notification, check *ERC20 token* balances, check *Ethereum* balances, get info on transactions and more. It can be used with any programming language that can send JSON encoded HTTP POST requests, which should be easy for every developer.

Current state of ERC20 token integrations

If **Tokengateway.io** is that helpful, how are token creators currently able to integrate their *ERC20 tokens* in web solutions and apps?

They currently have 3 options:

1. They are not integrating it and are doing all actions like sending tokens manually
2. They hire an expensive developer who will create a web3 based solution that requires every user to have the Metamask browser addon installed. Also the token creator now has to setup an Ethereum node that needs to be secured, updated and requires a lot of maintenance (and resources). Finally, they might even need to develop a new smart contract that has functions that can be called via web3, which costs a lot of money (Smart contract developers are the best earning developers at the moment).
3. They use a centralized API that will care about the blockchain and nodes part. With current solutions, this is possible with popular Tokens and cryptocurrencies. But what about the rest? They will get declined in the listing process or have to pay very high listing fees. And even if they manage to get listed, they have to pay a percentual fee for every transaction.

You see, it still isn't easy to work with *ERC20 tokens*, although there are over 200,000 different *ERC20 tokens* created already.

How does Tokengateway.io solve these problems?

1. **Tokengateway.io** allows to automate tasks like sending and receiving of *Tokens* and *Ethereum*.
2. Instead of the token creator, we are providing the **needed infrastructure** so the token creator can focus on his core business.
3. We are not charging a percentual fee for transactions. Also, we are not charging a listing fee for new tokens that want to use our solution – we don't even need to list new tokens as our API is designed to work with any *ERC20 compliant token* by default.

You may ask how we earn money if we don't charge percentual fees or listing fees. We will explain this on the next chapter…

Security of funds and data

In the crypto industry, **security** has a higher priority than in almost any other industry. Several *hacking attacks* in the past years have shown that even *big exchanges* can be hacked. All in all, we can say that no system is *100% secure*.

We still have a concept for ensuring security of funds. Any Ethereum address created on our system needs a password. Only when this password is provided and correct, transactions can be executed from the address. We do not store these passwords in plain text and also do not log them. Whenever a transaction is being requested, our API will check if the password, the Ethereum address and the API key is correct, before sending the transaction to our nodes, which will then verify the password again and sign the transaction. Also, we do not store private keys and passwords on our nodes. In addition, our nodes are not accessible from the internet.

User and API related data is stored in our fail-safe and redundant database cluster, reducing out times to a minimum. To reduce the loss of data, there are automatic and regular backups of all databases. KeyStore's are also back upped using a one-way live synchronization to an external backup server.

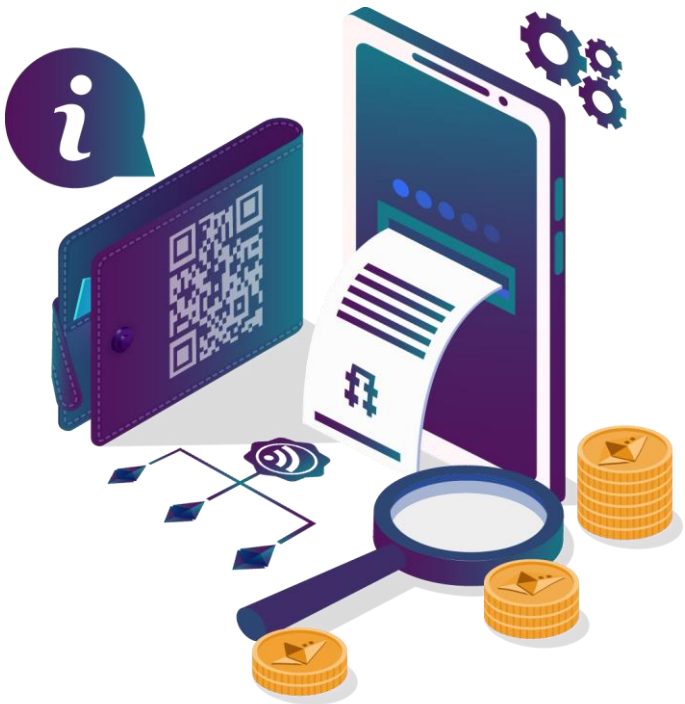
Monetarization

After doing some tests, we decided to use a monthly subscription system. Currently here are 3 kinds of requests that can be send to our API.

1. **Info Requests** – All requests that will return a value, like balance checks, getting the block number of the last mined block, getting info on transactions but also listing created Ethereum addresses and IPNs.
2. **Address Requests** – Requests relating to Ethereum addresses, like creating new addresses and deleting them.
3. **Transaction Requests** – Requests for sending transactions. This includes for example sending tokens, sending Ethereum and clearing Ethereum addresses (which is a function that calculates the total Ethereum balance and then forwards it to another address, more about that later)
4. **IPN/Subscription Requests** – These requests are for subscribing or unsubscribing to Ethereum addresses. You can either watch deposits of Ethereum or a specified ERC20 token.

For every plan we offer, there are different limits of requests you can use in total/monthly.

Info requests do have a monthly limit, which means after 1 month, this limit will be reset. Transaction requests also have a monthly limit. If you exceed the monthly limit, you need to upgrade to a bigger plan to continue using the API.



The addresses and IPN requests however have a total limit. This means the limit won't be reset every month. If you exceed the limits, you can either delete existing addresses/IPNs you don't need anymore or you have to upgrade your plan.

For an overview of the plans, including limits and pricing, please check out our pricing page. We are also happy to offer customized plans if the existing ones don't fit your needs. Just get in contact with us if you need a custom plan.

Available Blockchains

Using Tokengateway.io you can currently interact with 2 different blockchains:

1. **Ethereum Mainnet**

   The Ethereum blockchain is the one with the most blockchain projects as it allows to create decentralized applications on the chain, the so-called dApps. Also, the community and developers behind it are very active. Ethereum is also the second largest cryptocurrency after Bitcoin. Almost all tokens are running on this blockchain, which is why we are supporting this blockchain from the beginning.

2. **Ethereum Rinkeby**

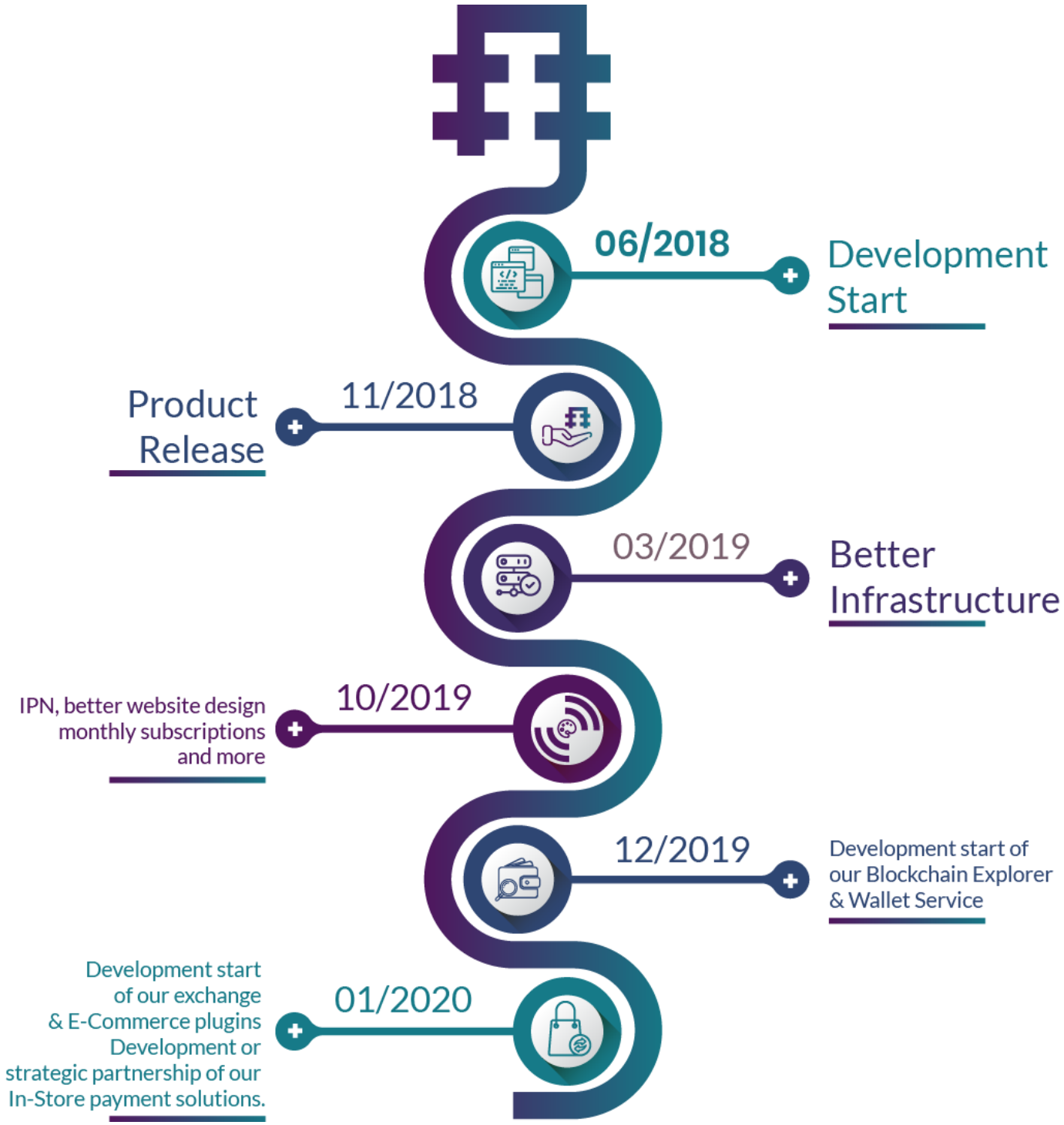
   The Ethereum Rinkeby blockchain is one of four blockchains to test Ethereum smart contracts and dApps, As the costs for transactions and deploying smart contracts on the Ethereum Mainnet can be expensive, most developers use one of the available test nets to test their applications without spending a single cent for network fees. We started to support this blockchain for exactly the same reasons: Testing our API, Tokens and more without the need to spend network fees.

Functions in Detail

We provide plenty of functions to interact with the Ethereum blockchain. Also, we are continuously adding new functions. You can always find the latest function overview and implementation details on our Documentation page.

1. **Info Requests**
   1.1. *getToken* - Returns information about a specific ERC20 token like name, symbol, decimal places and total supply.
   1.2. *getLastBlockNumber* - Returns the block number of the last mined ethereum block.
   1.3. *getGasPrice* - Returns the current gas price in GWEI.
   1.4. *getExchangeRate* - Returns the current Ethereum price in Euro or US Dollar.
   1.5. *getEthereumBalance* - Returns the ethereum balance of a given address.
   1.6. *getTokenBalance* - Returns the token balance of a given address.
   1.7. *getTransaction* - Returns information like confirmations, token contract address, amount, gas price and more of a given transaction.
2. **Address Requests**
   2.1. *newAddress* - Generates a new ethereum addresses you can use to send or receive funds.
   2.2. *deleteAddress* - Deletes an existing ethereum address. Be careful when using this function.
   2.3. *listAddresses* - Returns all ethereum addresses created with an account.
3. **Transaction Requests**
   3.1. *sendEthereum* - Sends ethereum from an address controlled by the account to a specified receiver address.
   3.2. *sendToken* - Sends ERC20 tokens from an address controlled by the account to a specified receiver address.
   3.3. *clearAddress* - Sends all available ethereum funds of an address to a specified receiver address.
4. **IPN/Subscription Requests**
   4.1. *subscribeAddress* - Creates a new subscription/IPN for the given address (and contractaddress). You will receive a notification to the given url every time a deposit is received.
   4.2. *UnsubscribeAddress* - Deletes an existing subscription/IPN for the given address (and contractaddress).
   4.3. *listSubscribedAddresses* - Returns all subscriptions/IPNs created with an account.

Roadmap


06/2018
Development Start
11/2018
Product Release
03/2019
Better Infrastructure
10/2019
IPN, better website design monthly subscriptions and more
12/2019
Development start of our Blockchain Explorer & Wallet Service
01/2020
Development start of our exchange & E-Commerce plugins Development or strategic partnership of our In-Store payment solutions.