



Squeezed Whitepaper

Executive Summary

The blockchain software industry is expanding quickly, given that blockchain related jobs are among the fastest growing in today's labor market, it is safe to assume that the demand for blockchain experts is also growing at a significant rate. The average income of a Blockchain developer (in US) has also increased by roughly 50% compared to that of the average software developer.

Vision

At Squeezers, we are creating a product stack that will help developers create dApps, and deploy them blazing fast. or region with just a simple click. We also offer a sandbox in which developers can test their applications and make rapid production deployments.

Squeezers applications are powered by microservices. Microservices are high scalable and have blockchain characteristics like block size limit and not being monolithic.

The Squeezers Platform will empower developers to compile and deploy blockchain applications in multiple stages directly from repositories similar GitHub, among other code repositories.

Background

In 2015, our founder and CEO, Nick Chisiu, came up with the idea to create a **framework** that would be able to develop, compile, and deploy traditional projects on microservices platforms. By providing consultancy services to various top 1000 US INC companies, he was able to implement the framework in production environments and sustain unlimited requests based on real user's demands.

In 2017, Nick joined **ConsenSys**, a top-five blockchain development company, as a blockchain consultant. By providing consultancy services and delving deeply into blockchain technology, he discovered a connection between microservices and blockchain, these two mixed together perfectly.

As a result the Squeezer Platform and the Squeezer Chainkit came into the picture, the other two main components of the Squeezer family.

At around the same time, Nick assumed the role of CEO and technology-lead at Squeezer and dedicated himself to the company full-time. His direction for the Squeezer Project is clear: deliver the roadmap and ensure that the development team creates robust, flawless components for blockchain integration.

The problem

Complexity

As a relatively new technology, blockchain development presents a number of challenges. In order to start developing a blockchain project, an individual needs to download all the blocks onto a machine's local storage (which for BTC represents 400 GB; for ETH – 100 GB; and for LTC – 200 GB approximately) which will eventually decrease the developer's velocity. Replicating this within the production environment represents another complex task; the developer now needs to create both a stable environment and containers that can replicate hundreds of gigabytes.

At a specific number of requests, that endpoint will eventually crash, because it is not designed to scale automatically, as the blockchain is. Of course, there are many possible solutions to such a problem, but it generally requires a lot of extra time and resources.

Onboarding software developers into the blockchain ecosystem is not that simple, because besides the coding skills, server side requirements are needed too, in order to setup blockchain node instances. In addition, strong typed languages like Solidity, C++, are additional points on the requirement list. All of these will narrow down the access to the developers workforce spectrum.

Connecting to the blockchain

Besides the auto-scalability issue, there is one more concern related to reading and writing data to the blockchain.

Blockchain data is stored in logs and each block consists of a log of transactions. A new bitcoin block for example, is generated every 20 minutes. Imagine that you need to retrieve all transactions for a specific wallet address. To do this, you would need to set a node and parse all available blockchain data, which requires a phenomenal amount of computing power. When dealing with sensitive data, it is best to find a solution that does all the heavy lifting for you, it's secure, and ensures that all data is accounted for.

The Solution

Squeezer Framework

One tool to build & deploy

The Squeezer Framework is an open source CLI for building and deploying decentralized applications using serverless functions. Instead of doing everything manually, you can create and configure resources with just a few commands, letting Squeezer do all the heavy lifting for you.

Squeezer Platform

Team Collaboration and Continuous Integration

Import a project from GitHub, share it with your team for collaboration, and start the development in seconds. Deployment of your dApp into the cloud is just a click away with the help of the Squeezer Platform.

Squeezer ChainKit

Warp Fast Solution for Blockchain Payments and Smart Contracts

Start accessing blockchain resources easily using the Squeezer Chainkit, and build serverless dApss. It's an agnostic solution to interact with any blockchain, so you don't have to deal with all the hassles of manually downloading the entire blockchain locally and setting up a dev environment.

ChainKit Features



Wallet Security

Chainkit empowers the end user to manage and store his own private credentials and by adding double wallet encryption.



Bidirectional Payment Channels

Get integration for both inbound and outbound blockchain transactions.



Blockchain agnostic

We help you access different blockchains or switch between them with only one line of code.



Smart contract secure access

Squeezer adds an extra layer on top of your smart contract to protect private methods and make it easier to access



Instant Chain Transactions

Transactions are made instantly without waiting the blockchain confirmations or other delay issues.

Wallet Double Encryption

Squeezer improves the way how a blockchain wallet is created, encrypted and stored. In order to get double encryption for a wallet Squeezer is using 2 actors (secret keys).

1. The first key is stored in to the dApp and is owned by the system admin
2. The second key is stored and owned by the end user (client)

In order to get the wallet decrypted and use it to trigger blockchain transactions, both access keys are required, so a possible attacker would need both keys for accessing the wallet, making it extremely secure.

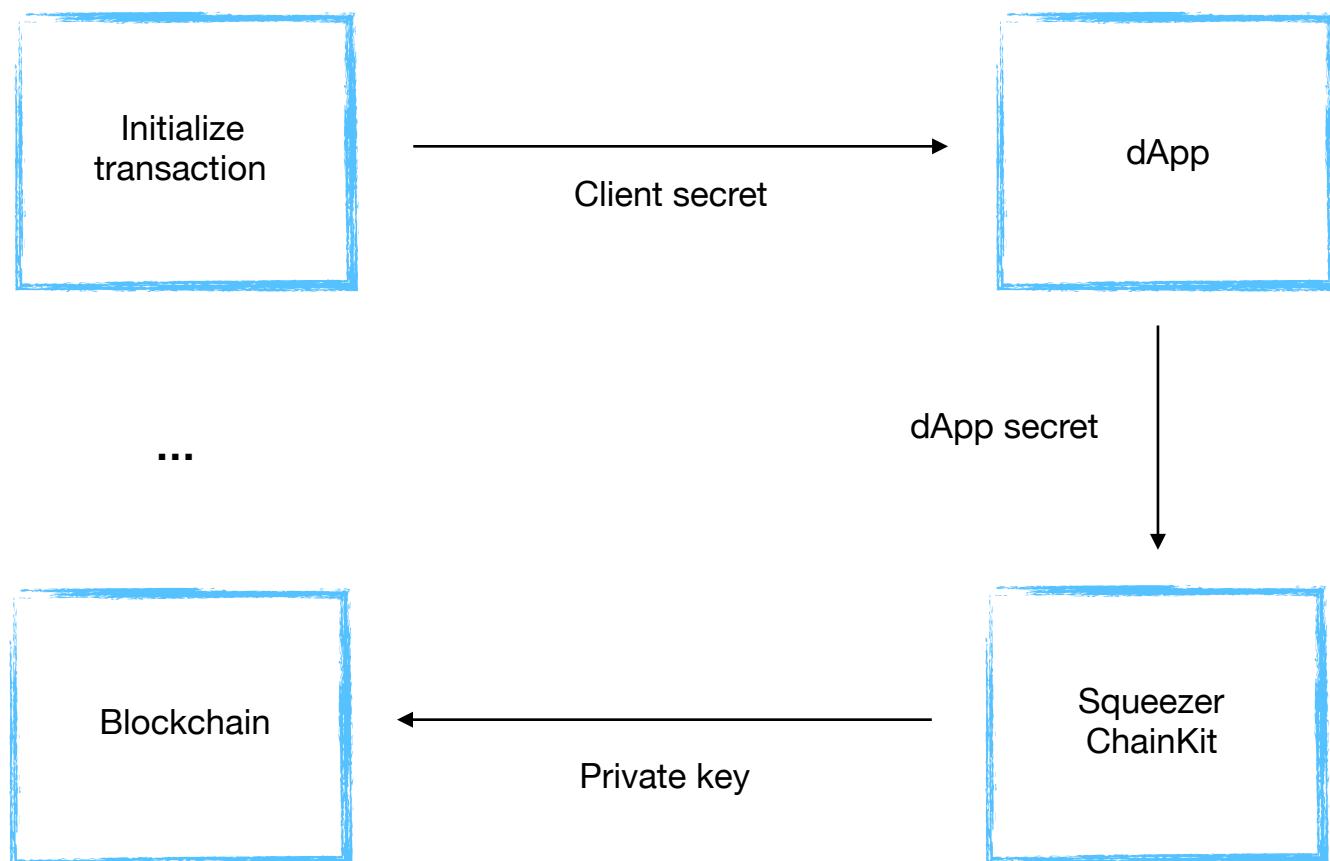


Fig 1.1

Instant On-Chain Transactions

A bitcoin transaction can take around < 10 minutes, Ethereum < 1 minute ... how does Squeezers manage to integrate instant on-chain transactions ?

Squeezers is created from scratch in such a way that it encourages a decentralized and secure environment, by empowering users to store the wallet data on their own. Still, even if the user is the final data possessor, Squeezers remains on the main stage to play a role when a new transaction is triggered. By being the only option for triggering a transaction and accessing wallet funds, Squeezers is able to record a transaction instantly, even before being sent to the blockchain for validation. To make sure that the real balance is inherited on the transaction Squeezers and to avoid **double spending attack** Squeezers will check the balance from its own ledger and from the blockchain, where both amounts will required to meet the criteria of being equal, otherwise an error will be thrown. Squeezers protects adds double spending attack support for blockchains by default, even if the blockchain doesn't natively support it.

The only way to pull out assets from a wallet created with Squeezers is by using Squeezers ChainKit, therefore the system will always know the real wallet balance which helps to make instant on-chain transactions from a wallet to another.

Squeezer Ledger (Distributed)

Squeezer ledger is the registry where we keep balances to check transactions if they meet the amount criteria for finalizing a new initiated transaction. In order to keep an immutable ledger and improve security on the transactions use **BigChainDB**

With high throughput, low latency, powerful query functionality, decentralized control, immutable data storage and built-in asset support, BigchainDB is like a database with blockchain characteristics.

BigchainDB allows developers and enterprise to deploy blockchain proof-of-concepts, platforms and applications with a blockchain database, supporting a wide range of industries and use cases.

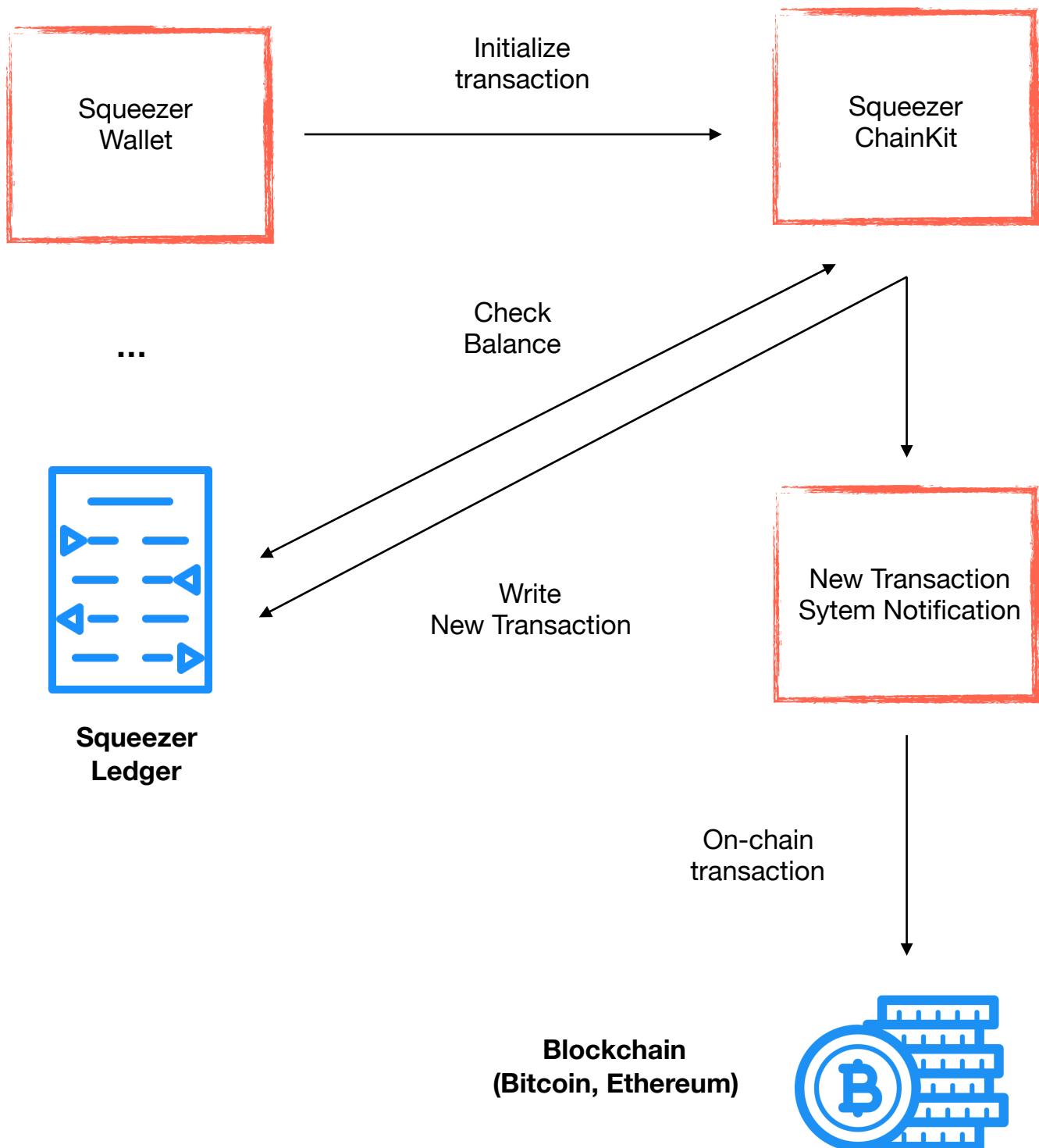


Fig 1.2

Transaction workflow

At a first glance the workflow doesn't look sophisticate, a user initializes a new transaction trough **ChainKit** which interrogates the ledger to see if there is enough balance to initiate the transaction, if yes we record it to the ledger and send it to the blockchain as **on-chain transaction**.

Transaction speed

Squeezer transactions are instantly on wallet to wallet transfer which we call **internal** transactions, even if the transactions are made on-chain. For other type of transactions that includes wallet withdraws or deposits we name them as **external** transactions. The balance from external deposits is available only when is blockchain confirmed, once is there it can be transferred instantly to other wallet.

Fees

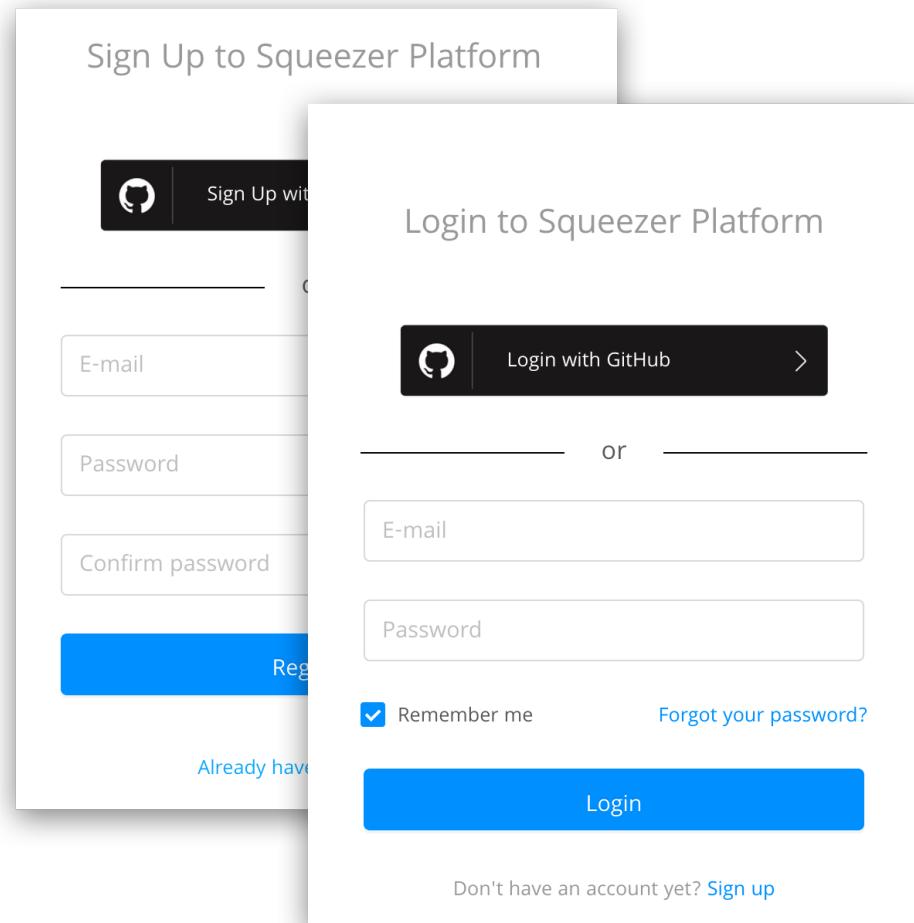
On-chain transactions are expensive mostly for **BTC** and **LTC**, still reasonable on most of the other chains. In order to satisfy the bitcoin community we bring Lighting Network, not for the speed as Squeezer already has it but for getting lower transaction fees.

Squeezer Platform

The Squeezer Platform is the user interface part of the stack. It offers the ability to import your squeezer project which you've developed using the Squeezer Framework CLI and deploy it in the cloud.

Login

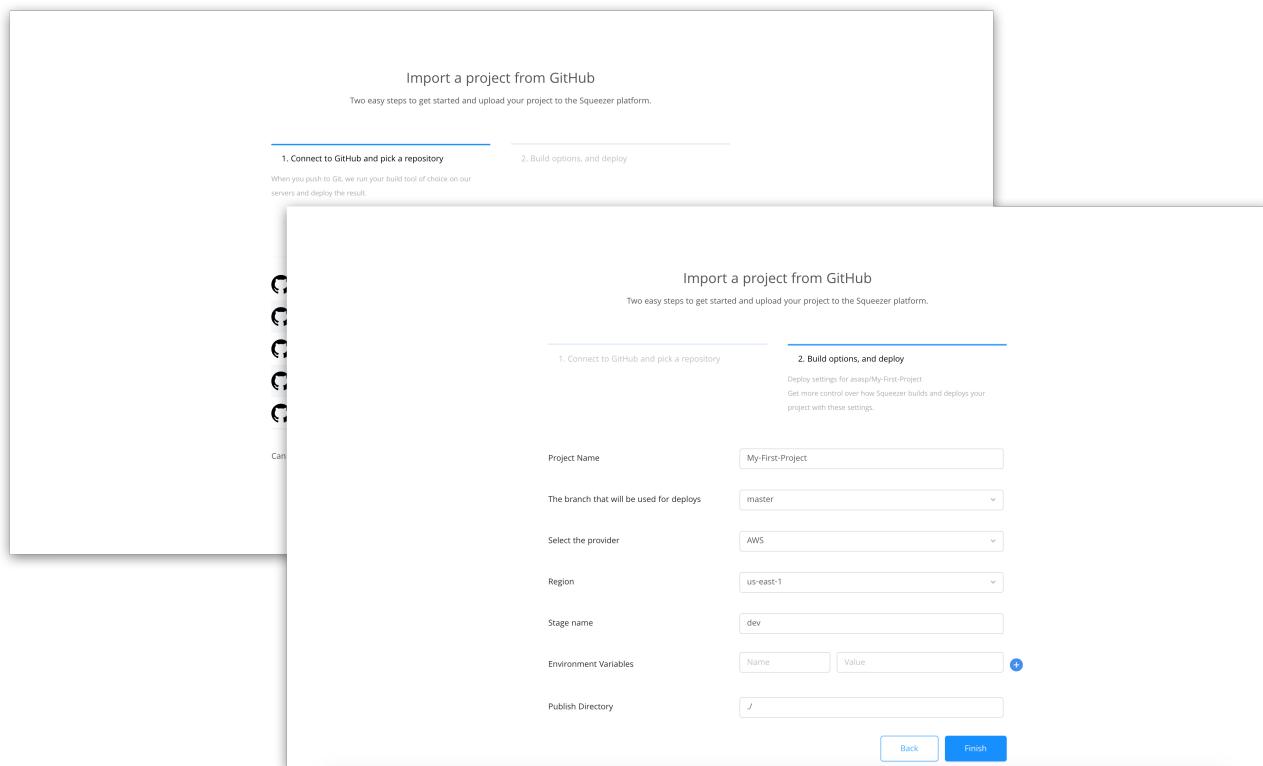
Login or Sign Up to the Squeezer Platform can be done using a GitHub account, or using generic credentials, e-mail and password.



Importing A Project

You can import a squeezer project from your GitHub account, in just 2 steps:

- After a successful GitHub authentication go to the import screen by clicking on the import project button/link.
- On the next screen of the import project, you can configure your deploy settings such as project name, the branch you want to deploy, provider, stage name, environment variables and the publish directory.



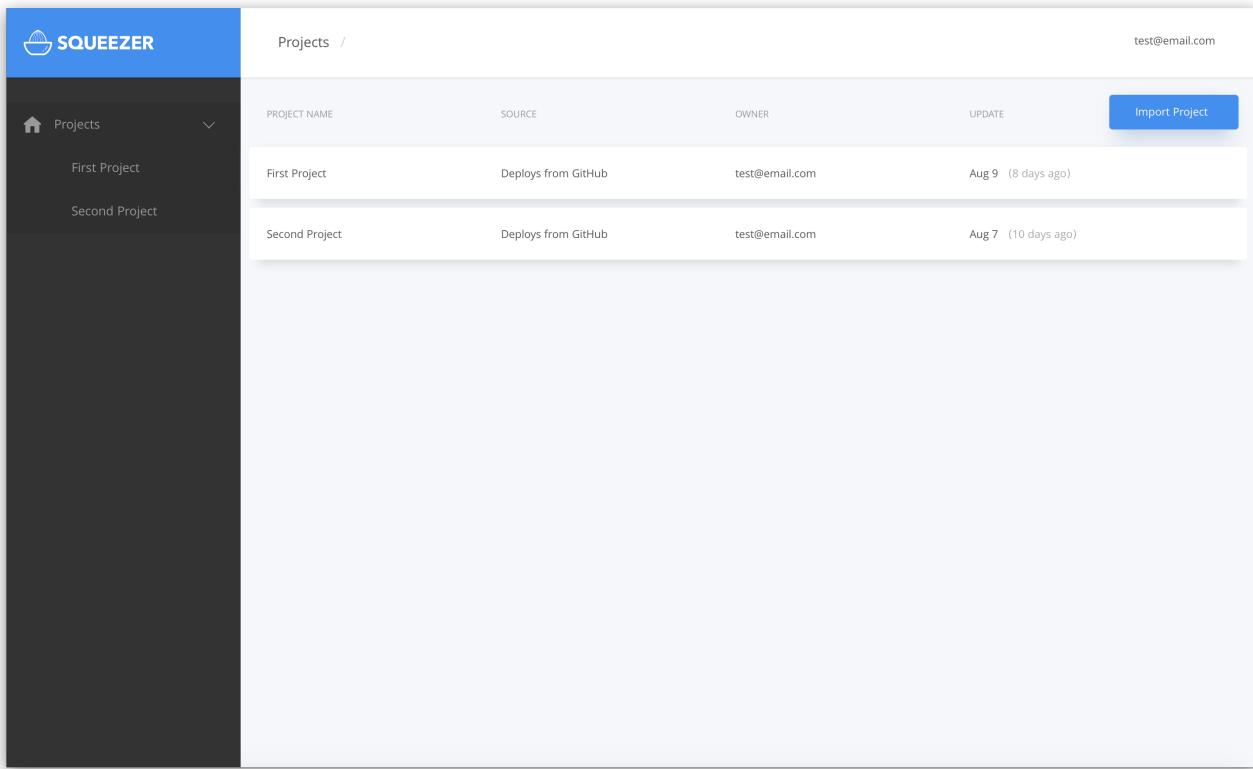
The screenshot shows the 'Import a project from GitHub' interface. It consists of two main sections: '1. Connect to GitHub and pick a repository' and '2. Build options, and deploy'.
Step 1: Connect to GitHub and pick a repository
This section includes a note: 'When you push to Git, we run your build tool of choice on our servers and deploy the result.'
Step 2: Build options, and deploy
This section includes a note: 'Deploy settings for easyMyFirstProject' and 'Get more control over how Squeezers builds and deploys your project with these settings.'
The configuration fields are:

- Project Name: My-First-Project
- The branch that will be used for deploys: master
- Select the provider: AWS
- Region: us-east-1
- Stage name: dev
- Environment Variables: A table with columns 'Name' and 'Value' with a '+' button to add more rows.
- Publish Directory: /

At the bottom are 'Back' and 'Finish' buttons.

Project Dashboard

The Project Dashboard is the place where you have an overview of all your imported projects with details such as the source of the import, the owner and the date it was last updated.



The screenshot shows the Project Dashboard interface. On the left is a sidebar with a blue header containing the SQUEEZER logo. Below the header, the sidebar has a 'Projects' section with a house icon and dropdown arrow, listing 'First Project' and 'Second Project'. The main area is titled 'Projects /' and shows a table of imported projects. The table has columns: PROJECT NAME, SOURCE, OWNER, and UPDATE. It contains two rows:

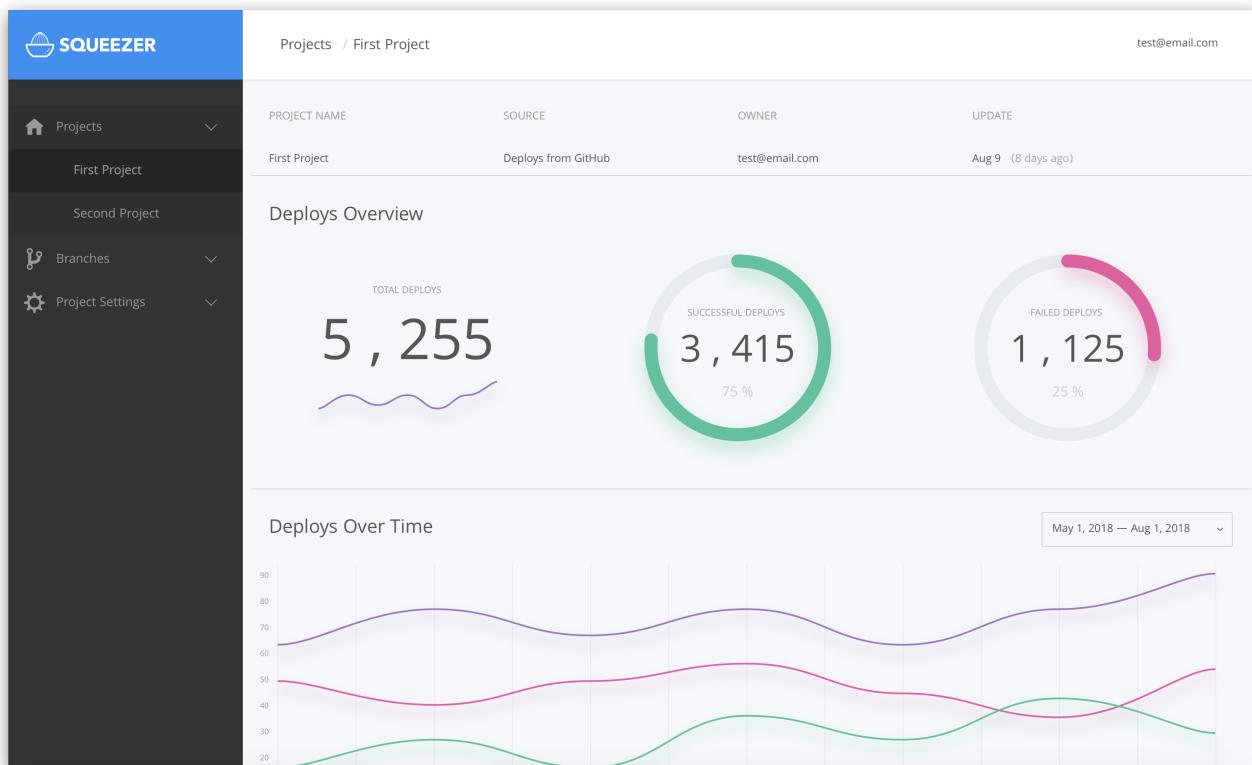
| PROJECT NAME | SOURCE | OWNER | UPDATE |
|----------------|---------------------|----------------|---------------------|
| First Project | Deploys from GitHub | test@email.com | Aug 9 (8 days ago) |
| Second Project | Deploys from GitHub | test@email.com | Aug 7 (10 days ago) |

A blue 'Import Project' button is located at the top right of the table area. At the top right of the main dashboard area, the email 'test@email.com' is displayed.

There's also an Import Project button which allows you to import any project from your GitHub account.

Viewing The Project Details

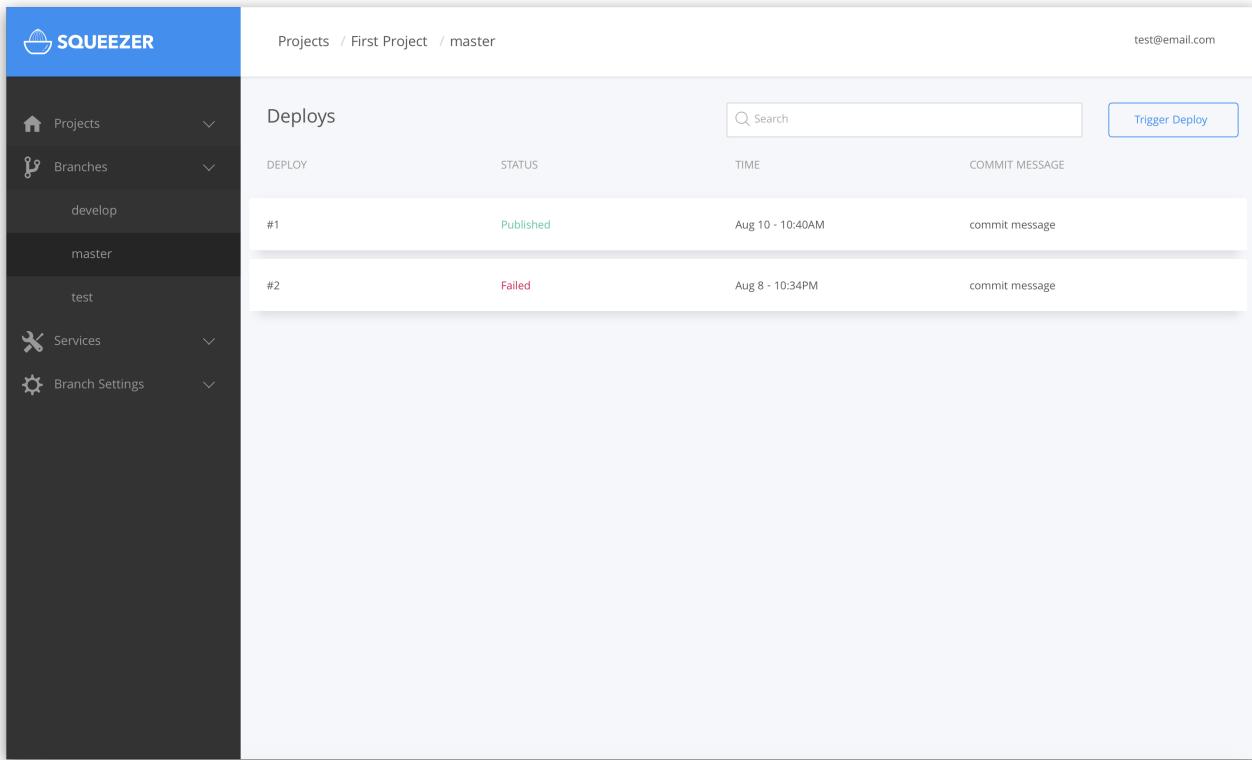
After clicking on the desired project from the Project Dashboard, the project details screen opens up. Here you can see the total number of deploys this project currently has, and you can also select to view deploys over time



Since a project can have multiple branches imported, the above deploys represent the total number of deploys per all the branches of a given project.

Branch Deploys

After selecting a project, you can choose any branch that was imported, to see deploys per branch, and details like the status of the deploy, last commits, and time of the deploy.



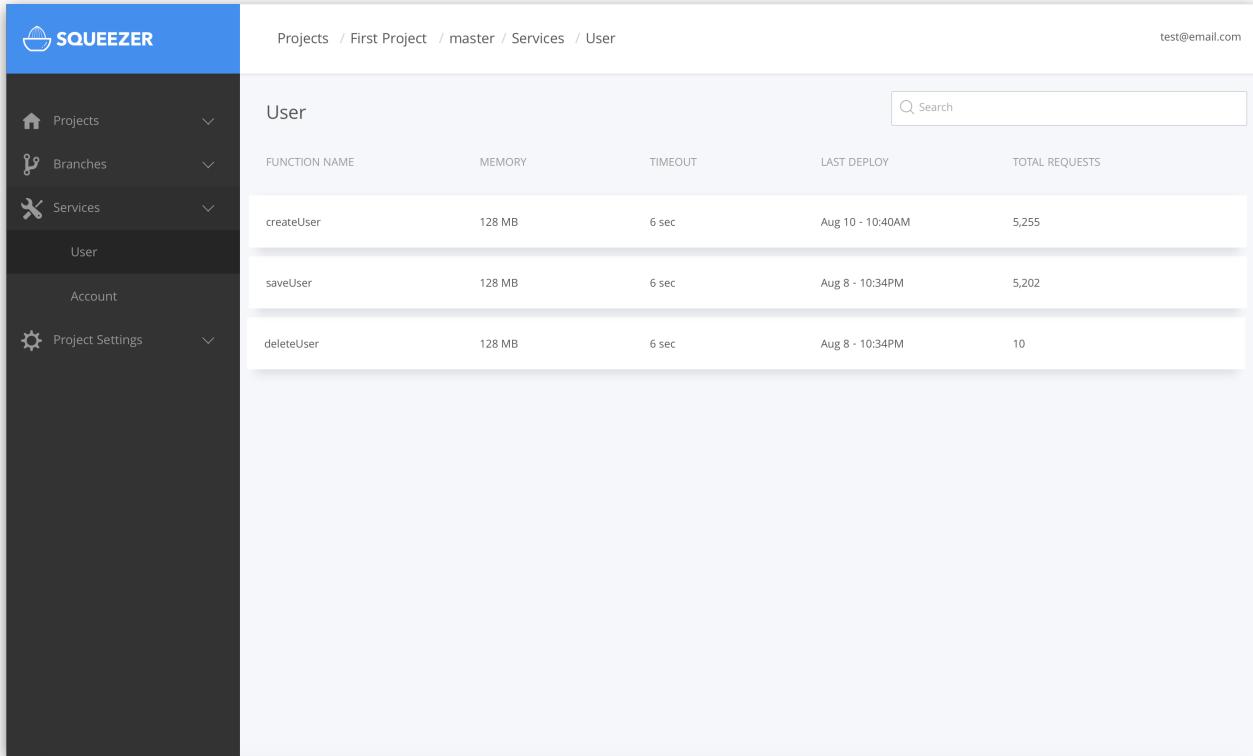
The screenshot shows the SQUEEZER web interface. On the left is a dark sidebar with a blue header containing the SQUEEZER logo. The sidebar has several sections: 'Projects' (with a house icon), 'Branches' (with a branch icon), 'develop', 'master' (which is highlighted in light blue), 'test', 'Services' (with a wrench icon), and 'Branch Settings' (with a gear icon). The main area has a white header with 'Projects / First Project / master'. On the right, there's an email address 'test@email.com'. Below the header is a search bar and a 'Trigger Deploy' button. The main content area is titled 'Deploys' and contains a table with two rows. The columns are labeled 'DEPLOY', 'STATUS', 'TIME', and 'COMMIT MESSAGE'. Row #1 shows 'Published' status, 'Aug 10 - 10:40AM', and 'commit message'. Row #2 shows 'Failed' status, 'Aug 8 - 10:34PM', and 'commit message'.

| DEPLOY | STATUS | TIME | COMMIT MESSAGE |
|--------|-----------|------------------|----------------|
| #1 | Published | Aug 10 - 10:40AM | commit message |
| #2 | Failed | Aug 8 - 10:34PM | commit message |

If a deploy fails for no apparent reason, you can always try to trigger the deploy manually, without the need to import the project again.

Services Overview

In the Services dropdown from the left navigation menu, you can see the services and functions that your project contains, along with other information, such as what amount of memory each function has allocated to it, the timeout, last deploy date and total number of requests.



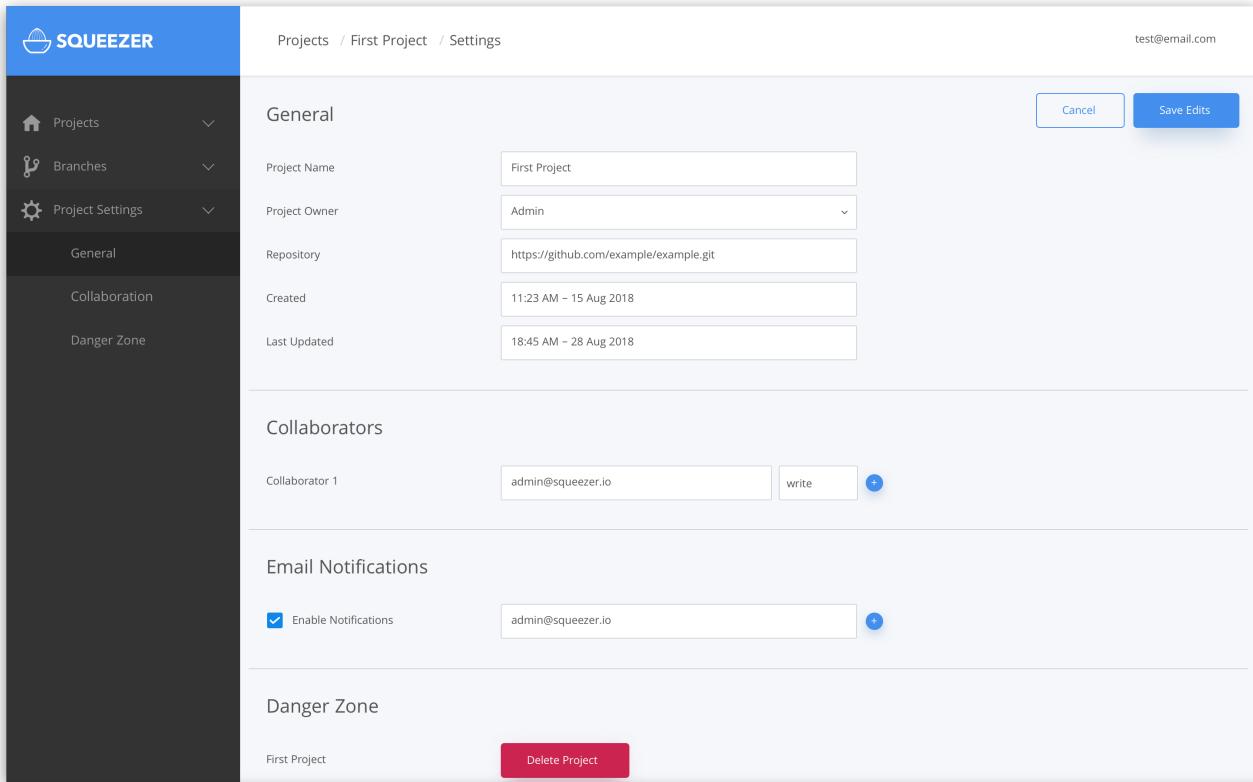
The screenshot shows the SQUEEZER interface. The left sidebar has a blue header with the SQUEEZER logo and navigation items: Projects, Branches, Services (selected), User, Account, and Project Settings. The main content area shows the 'User' service with a search bar. A table lists three functions:

| FUNCTION NAME | MEMORY | TIMEOUT | LAST DEPLOY | TOTAL REQUESTS |
|---------------|--------|---------|------------------|----------------|
| createUser | 128 MB | 6 sec | Aug 10 - 10:40AM | 5,255 |
| saveUser | 128 MB | 6 sec | Aug 8 - 10:34PM | 5,202 |
| deleteUser | 128 MB | 6 sec | Aug 8 - 10:34PM | 10 |

Project Settings

Project Settings will show you details regarding the owner, the name of the repository that was imported from GitHub and the import date.

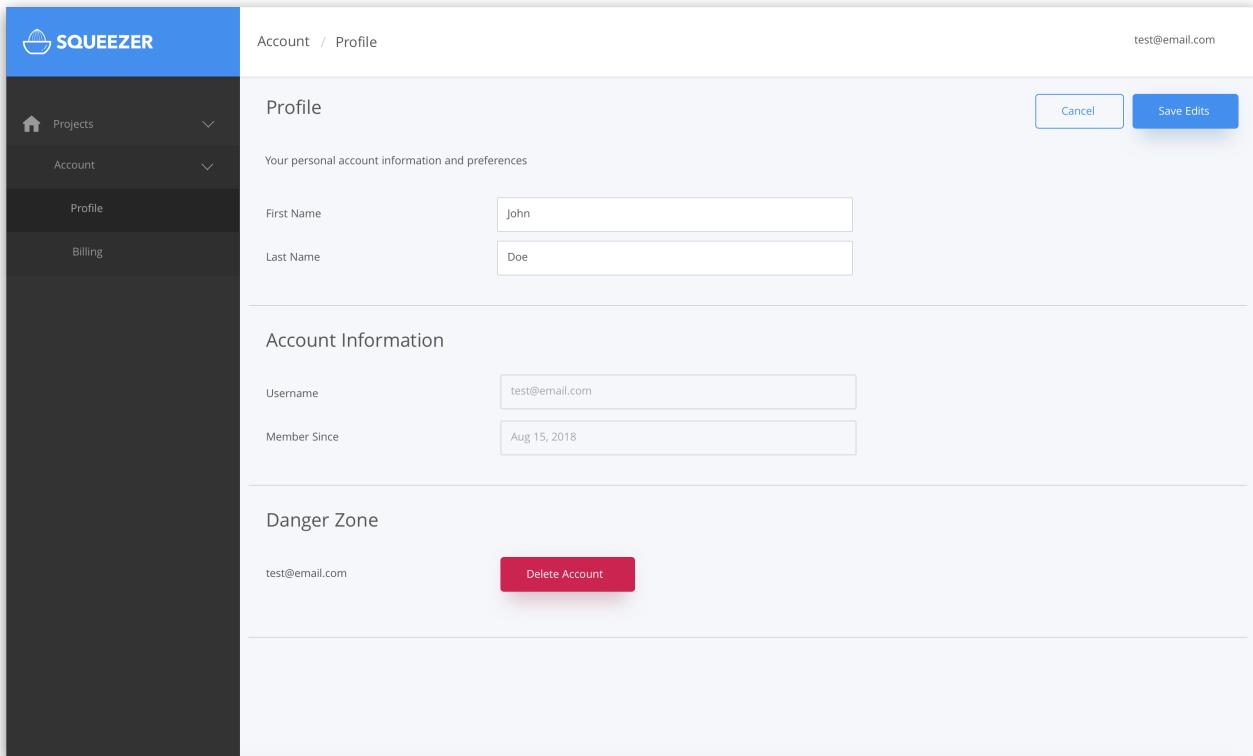
Here you can add collaborators, so you can work in teams on your project, and also enable notifications to get an e-mail each time a deploy is made or any other activity regarding your project.



The screenshot shows the 'Project Settings' page for a project named 'First Project'. The left sidebar has a dark theme with white icons and text, showing 'Projects', 'Branches', 'Project Settings' (selected), 'General', 'Collaboration', and 'Danger Zone'. The main area has a light background. At the top right, it shows the email 'test@email.com'. The 'General' section contains fields for 'Project Name' (set to 'First Project'), 'Project Owner' (set to 'Admin'), 'Repository' (set to 'https://github.com/example/example.git'), 'Created' (set to '11:23 AM - 15 Aug 2018'), and 'Last Updated' (set to '18:45 AM - 28 Aug 2018'). Below this is a 'Collaborators' section with a table row for 'Collaborator 1' (email 'admin@squeezero.io', permission 'write') with a '+' button. Under 'Email Notifications', there is a checked checkbox for 'Enable Notifications' and a text input field containing 'admin@squeezero.io' with a '+' button. The 'Danger Zone' section at the bottom has a red button labeled 'Delete Project'.

Account Settings

In this section of the platform, you can edit your profile information, and delete your account. Your username, by default, will be the e-mail address you've signed up with, and cannot be edited. This section will also show the date your account was created on.



The screenshot shows the 'Profile' section of the SQUEEZER account settings. The left sidebar has 'Projects' and 'Account' dropdowns, with 'Profile' selected. The main area shows 'Profile' information: First Name (John) and Last Name (Doe). Below that is 'Account Information' with Username (test@email.com) and Member Since (Aug 15, 2018). At the bottom is a 'Danger Zone' with the email test@email.com and a red 'Delete Account' button.

Account / Profile

Profile

Your personal account information and preferences

First Name: John

Last Name: Doe

Account Information

Username: test@email.com

Member Since: Aug 15, 2018

Danger Zone

test@email.com

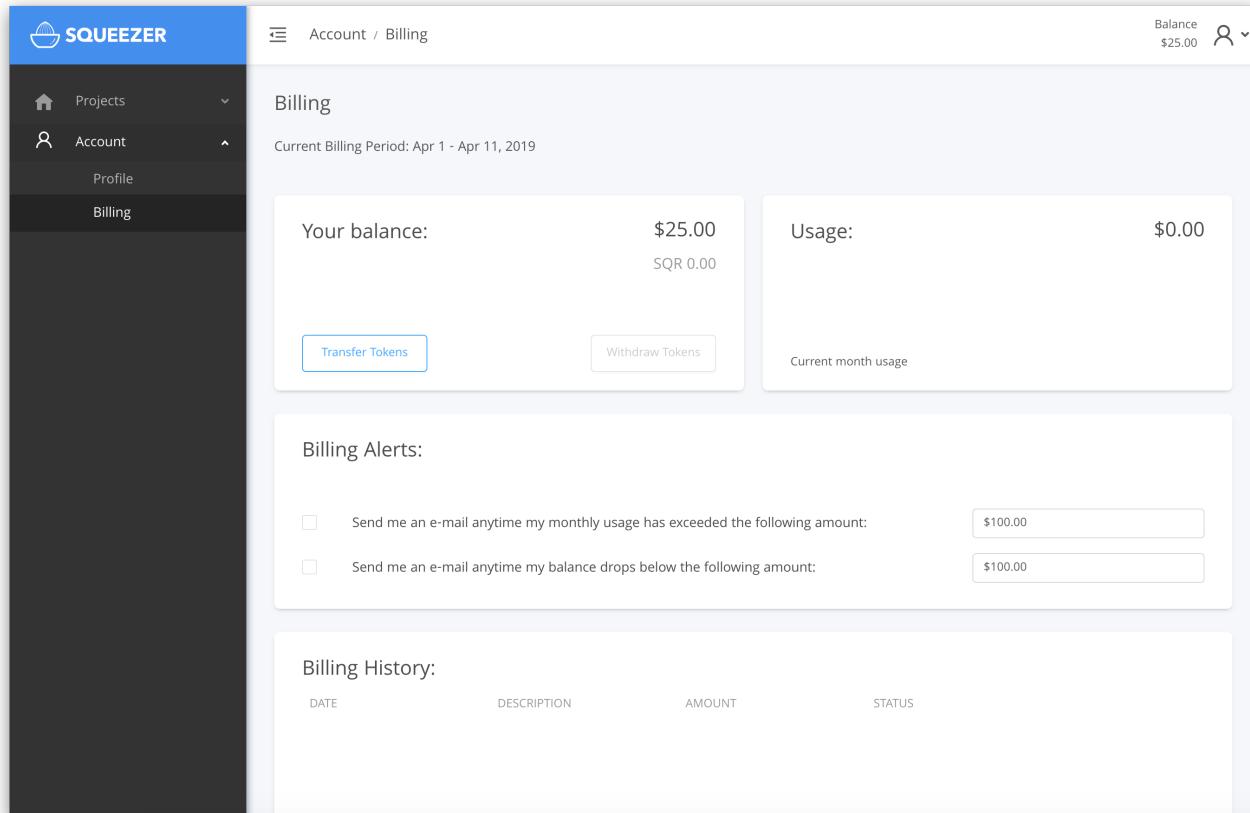
Delete Account

Billing

Billing is yet another important part of the Squeez Platform. Here you can view details about your balance, which can be in USD or SQR depending on the top up method, and the usage for the current month. You have the possibility to transfer tokens to any other Squeez user.

You can also setup billing alerts for when the monthly usage exceeds or drops under a certain amount.

You will also have a billing history where you can view all of your invoices, and export them in pdf or excel format.



The screenshot shows the Squeez Platform's Billing interface. On the left is a dark sidebar with a blue header bar containing the Squeez logo and the word "SQUEEZER". The sidebar has four items: "Projects", "Account" (which is expanded to show "Profile" and "Billing"), and two other collapsed items. The main content area has a white header bar with the text "Account / Billing" and "Balance \$25.00" along with a user icon. Below this is a section titled "Billing" with the subtitle "Current Billing Period: Apr 1 - Apr 11, 2019". This section contains two boxes: one for "Your balance:" showing "\$25.00" and "SQR 0.00", and another for "Usage:" showing "\$0.00" and "Current month usage". There are "Transfer Tokens" and "Withdraw Tokens" buttons. Below this is a "Billing Alerts:" section with two checkboxes: "Send me an e-mail anytime my monthly usage has exceeded the following amount:" with a field containing "\$100.00", and "Send me an e-mail anytime my balance drops below the following amount:" with a field containing "\$100.00". At the bottom is a "Billing History:" section with a table header row showing columns for DATE, DESCRIPTION, AMOUNT, and STATUS.

| DATE | DESCRIPTION | AMOUNT | STATUS |
|------|-------------|--------|--------|
|------|-------------|--------|--------|

Competitive advantages

A few microservice frameworks currently exist, but none of them are blockchain oriented or have given any indication in their roadmap that they intend to facilitate blockchain integration in the future, so Squeezer remains the only microservice based tool out there that is blockchain friendly.

Another advantage of the Squeezer project is that it offers a complete set of tools that helps developers build decentralized applications from start to finish.

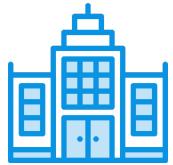
The **Squeezer Frameworks** offers an open source command line interface to build the services and functions for your app's backend.

with the help of the **Squeezer Chainkit** you can access blockchain as a service and interact with blockchain resources easily.

Importing your project from GitHub into the **Squeezer Platform** will help you then deploy, maintain, monitor, and debug your code without any hassles whatsoever.

Main features of Squeezers

Zero administration



Deploy your code without any lengthy set-up beforehand or anything to manage afterward.

Blockchain



A truly revolutionary framework, Squeezers is the first platform that merges the power of microservices with the immutability of blockchain technology.

Development Climate



Simulate the cloud provider environment on your local machine. No need to deploy code at every iteration. Speed up the entire development cycle by 10x.

Pay-per-use

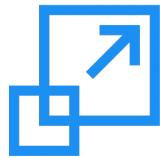


Function-as-a-Service (FaaS) computing and managed services charged based on usage rather than pre-provisioned capacity. You can utilize all your resources without paying a cent for idle time.



Token

The Squeezer Token (SQR) is the core utility used by developers to build and deploy apps on the Squeezer Platform.



Scalability

Let your service providers manage the scaling challenges. No need to set alerts or write scripts to scale up or down. Have absolute peace of mind during periods of high or low traffic.

Business model

Summary

Squeezer is designed as a middleware between blockchains and traditional software. There are so many companies looking for solutions to integrate blockchain into their infrastructure, such as gaming companies, social network companies, etc. The downside is that blockchain developers are extremely rare resources, and their compensations are too high. Those who are capable of leading a development team can make 80k USD a month in Shanghai. A blockchain developer in Silicon Valley is making 250-300k USD a year. Even so, it is still challenging to hire these talents, and gathering a team is a harder task.

We have developed Squeezers to be an easy solution for such integrations which can fill this gap having a huge market potential. It empowers companies and developers to integrate modern blockchain technologies into their products in an efficient and cost effective manner. You can think of squeezer as the gateway for blockchain technology mass adoption.

Squeezers will be designed and implemented easily to integrate with traditional software suites such as SAP, Oracle, Microsoft, AWS, etc. It will connect with the most advanced blockchain technologies in the backend, such as ZIL, XML, EOS, FAB, LoomX, Moac, etc. Squeezers is sophisticated and well designed. It takes care of all the manual tedious work such as server preparations, environment setup, and presents only simple, easy to use and solid built APIs to the end users.

Business and technical consulting firms are looking for opportunities to dive into the blockchain world. Squeezers is designed to be a cost efficient and accessible tool for accessing blockchain resources, so that businesses can integrate it easier and faster. Usually the first players in the game take the bigger slices of the pie. At this moment Squeezers is the only blockchain integration tool on the market that works with every blockchain out there, making it the biggest player in this business segment.

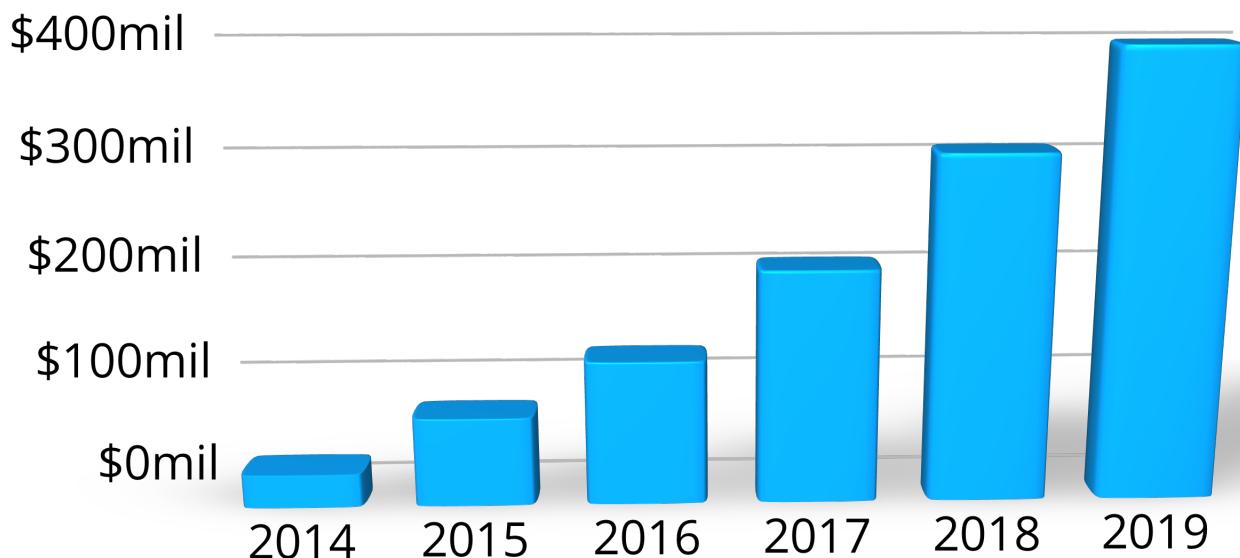
The SQR token is used in the Squeezers Platform to pay for services such as deployments, inbound and outbound transactions or additional team members to collaborate on your project.

Blockchain consultancy

The number of companies implementing blockchain technology has increased significantly across all sectors, and the demand for blockchain software developers is growing in response to this increase.

The following chart displays the amount spent by the banking sector on blockchain technology over the past five years. When you add in the amounts that governments, corporations, and other vendors have spent on blockchain, the total will be approximately **five billion USD** in 2017.

Estimated spendings on blockchain tech for the banking sector



Source: ATLAS | Data: Aite Group

Market

Our goal is to ensure that our clients can harness the full potential of cloud providers, blockchain technology, and the Squeezers software stack by providing them with our professional consultancy services. By offering the ultimate technology stack to our clients, we can streamline the integration of blockchain technology into their current enterprise systems.



1

Exchanges

The current cryptocurrency exchanges are experiencing difficulties caused by large transaction volumes over short periods of time. It is no secret that the largest exchanges have scalability issues, and they therefore have to block new user registrations from time to time. Squeezers offers a solution by integrating microservices into the core of transactional systems.

2

Banks and financial institutions

Banks are certainly interested in blockchain technology. However, most of the financial institutions we have surveyed are still in the early stages of adoption, with about three-quarters either involved in outlining a proof-of-concept, formulating their blockchain strategy, or discussing the technology at an even more preliminary stage. Squeezers represents the best tool to streamline workflows and deliver relevant insights to banks about how transactions are processed through the blockchain ecosystem.

Token Usage

Summary

Squeezer will be similar to PayPal, but for blockchain transactions, it will provide real time transactions support on different blockchains, the quickest integration at the moment in the industry (Squeezer Chainkit). This is not a merchant or a similar platform, we offer an end-to-end user integration, so we don't bridge transactions, the user will interact directly with the blockchain resources. Squeezer is the first set of tools that does this, and the market for such services is huge, growing day by day. This software stack is built having in mind the support for a very high number of transaction requests backed by microservices. Another great feature is that you can access smart contract methods directly using the Squeezer Chainkit. You can also build a voting system dApp in minutes.



Squeezer is about the simplicity, scalability & the velocity offered to the developer in a friendly manner to build dApps. The modern dApp requires blockchain transactions support and smart contracts, we got them both, plus we offer you the rest of the technology stack to build the dApp

Each blockchain successful transaction that goes through our platform (in/out) will be billed on a fixed rated on a pay-as-you-go tier. Additionally we will have standard subscriptions to host dApps + monthly support services package (similar AWS). All the consultancy services will be payed with SQR token. SQR token will be the only payment method available on the website.

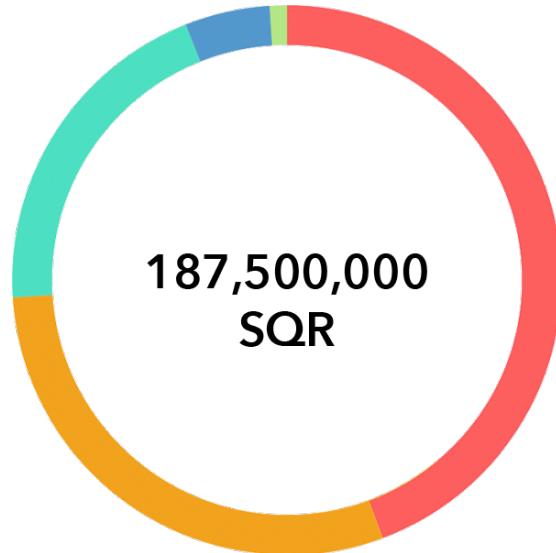
Payment type - **PAY AS YOU GO**

|  | | | | |
|---|---|---|--|---|
| | Tier 1 | Tier 2 | Tier 3 | Tier 4 |
| Outbound Transactions | 1-10000 \$0.05 (0.25 SQR) / unit | 10000 - 50000 \$0.04 (0.20 SQR) / unit | 50000 - 500000 \$0.03 (0.15 SQR) / unit | 500000+ \$0.02 (0.10 SQR) / unit |
| Inbound Transactions | 1 - 10000 \$0.04 (0.20 SQR) / unit | 10000 - 50000 \$0.03 (0.15 SQR) / unit | 50000 - 500000 \$0.02 (0.10 SQR) / unit | 500000+ \$0.01 (0.05 SQR) / unit |
| Deployments | 50 - 100 \$0.05 (0.25 SQR) / unit | 50 - 250 \$0.04 (0.2 SQR) / unit | 250 - 500 \$0.03 (0.15 SQR) / unit | 500+ \$0.02 (0.1 SQR) / unit |
| Team Members | 1 - 5 \$2.4 (12 SQR) / unit | 5 - 20 \$2 (10 SQR) / unit | 20 - 50 \$1.8 (9 SQR) / unit | 50+ \$1.4 (7 SQR) / unit |

Token Economics

| | |
|----------------------------|----------------------|
| Total Token Supply | 187,500,000 SQR |
| Total Token Sale Supply | 40% (75,000,000 SQR) |
| Private Sale Supply | 24% (45,000,000 SQR) |
| Public Sale Supply | 16% (30,000,000 SQR) |
| Initial Circulating Supply | 27% (50,791,761 SQR) |
| Public Sale Token Price | 1 SQR = 0.20 USD |
| Private Sale Token Price | 1 SQR = 0.17 USD |
| Public Sale Vesting Period | None |
| Token Type | ERC-20 |

Token Distribution



Token sale:

75,000,000 SQR (40%)

At the end of the token sale,
all unsold tokens will be burned.

The Squeezer Platform:

56,250,000 SQR (30%)

The reserve required to ensure the
operation of the platform. Locked in smart
contract with sale restrictions for 12
months.

Team:

37,500,000 SQR (20%) Locked in smart

contract with sale restrictions for 24
months.

Advisors:

9,375,000 SQR (5%)

Sale restrictions for 12 months

Marketing & partners:

9,375,000 SQR (5%)

Sale restrictions for 12 months

Token Proceeds

● Technical development: 45%

● Non-technical staff: 30%

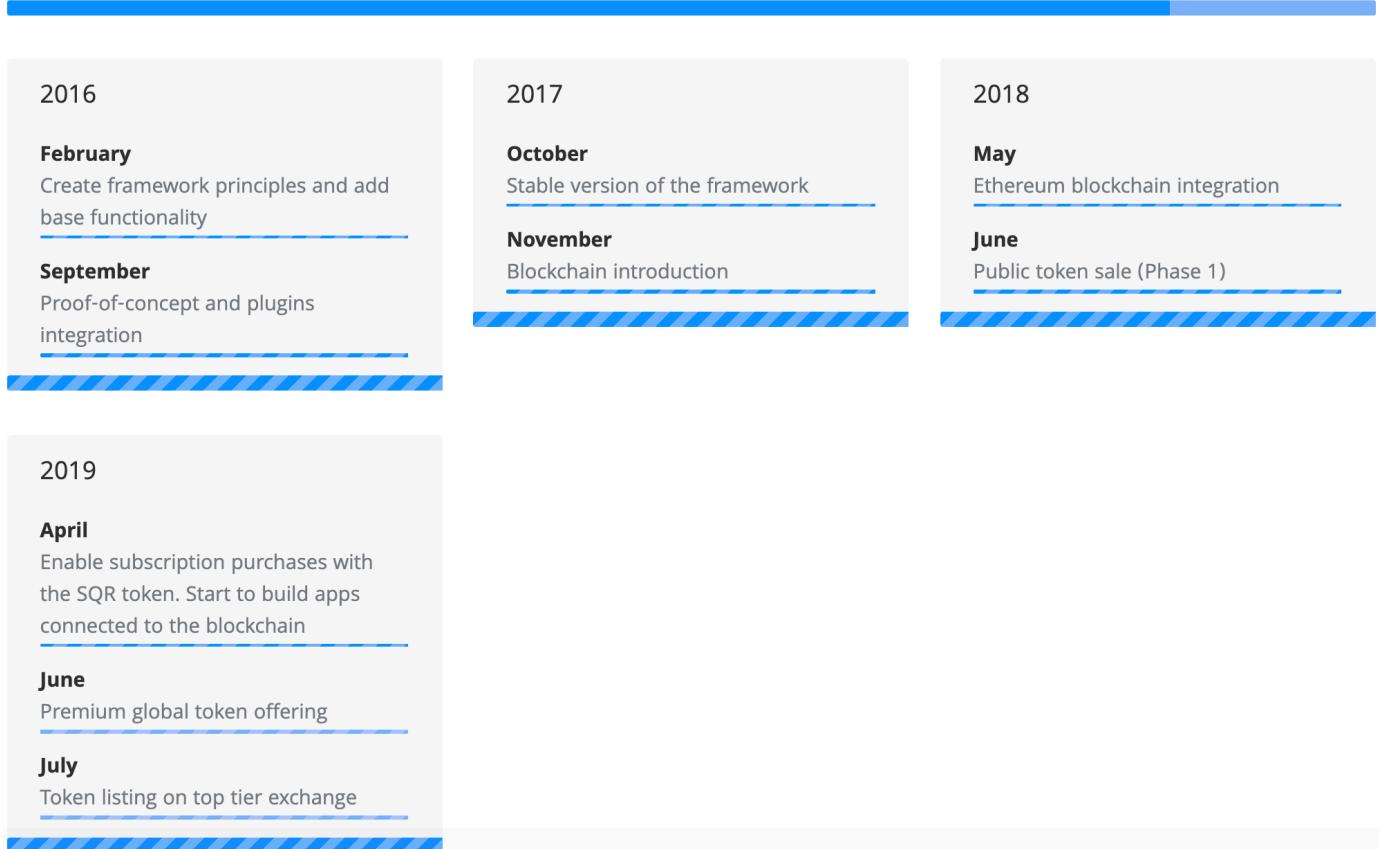
● Marketing expenses: 10%

● Infrastructure expenses: 8%

● Other operating expenses: 7%



Roadmap



Legal

General

The Squeezers token does not legally qualify as a security, since it does not give any rights to dividends or interests. The sale of Squeezers tokens is immutable and non-refundable. Squeezers tokens are not shares and do not give any right to participate in the general meeting of **Golden Data INC.** Squeezers tokens are intended to be used to buy application subscriptions and services on the Squeezers Platform. Any entity that buys Squeezers tokens expressly agrees and represents that she/he has carefully reviewed this white paper and fully understands the risks, costs, and benefits associated with the acquisition of Squeezers tokens.

Knowledge

The buyer of Squeezers tokens undertakes that she/he understands this white paper and has a minimum experience of cryptocurrency, blockchain systems, and services, and that she/he fully understands the risks associated with the crowdsale campaign as well as the workflow related to the use of cryptocurrencies (e.g. **secure storage**). Squeezers shall not be responsible for any loss of Squeezers tokens or situations that make it impossible to access Squeezers tokens, which may result from any trigger or from electronic hacking.

Disclaimer

This white paper should not be considered as an invitation for investment. There is no relation between the white paper and security in any jurisdiction. Trading the Squeezers tokens will not change the default legal qualification of SQR tokens, which always remains as a utility.

All content in the white paper is designed for general information purposes only and Golden Data INC does not provide any warranty as to the accuracy and completeness of this information. Golden Data INC is not to be considered an advisor in any financial, taxation, or legal objectives. Buying Squeezers tokens shall not grant any right or influence to the Buyers over Golden Data INC's organization and governance. Regulatory authorities are carefully auditing businesses and operations associated with cryptocurrencies around the globe. Accordingly, regulatory laws, investigations, or actions may affect Golden Data INC's business and even limit or prohibit it from developing its operations in the future. Any person undertaking to acquire Squeezers tokens must acquaint themselves first with Golden Data INC's business model. This white paper may change or need to be modified because of new rules and compliance requirements. In such a case, buyers and anyone else in possession of Squeezers tokens must acknowledge and understand that neither Golden Data INC nor any of its affiliates shall be held liable for any direct or indirect damage or loss to the buyer. While Golden Data INC will strive to follow the roadmap and build the platform, Squeezers token buyers acknowledge and understand that Golden Data INC does not provide any guarantees to the accomplishment of that goal.

KYC/AML

Participants who wish to purchase tokens will need to pass the KYC/AML verification first. We keep KYC/AML verifications privately and securely stored on a cloud server using **SUM & Substance** service.

Warranty

By contributing to the crowdsale campaign, the buyer agrees to the above and in particular, they represent and warrant that they:

- have read carefully the terms and conditions included in the white paper; agree to their full implications and accept to be legally bound by them;
- are authorized and are fully empowered to buy Squeezers tokens according to the laws/rules that apply in their domicile and jurisdiction;
- are neither a US, China, Canada and Hong Kong citizen or resident;
- live in a jurisdiction which permits Golden Data INC to sell Squeezers tokens through a crowdsale without requiring any additional authorization;
- are familiar with all related regulations in the public/private jurisdiction in which they are located and that acquiring cryptographic tokens in that jurisdiction is not restricted, prohibited, or subject to additional enforcements;
- will not use the crowdsale campaign for any illegal operation, including but not limited to financing of terrorism or money laundering;
- have sufficient knowledge about the specifications of the cryptographic tokens and have the minimal experience with, and functional behavior understanding of, the usage and dealings of cryptographic tokens and currencies and blockchain-based systems and services;

- buy Squeezers tokens because they wish to trade it or build blockchain apps in the cloud in future.

Governing law

Any dispute or issue arising from or under the crowdsale campaign shall be resolved/finalized in compliance with the Belize rules for IBC (as per the [IBC Act](#).)

Cloud Providers

The Squeezers Framework does not have any legal or partner qualification with the specified cloud providers. Therefore, there is no association or relationship implied between Squeezers token sales and the named cloud providers. Cloud providers are mentioned only for technical purposes and not for marketing purposes under any circumstances.