



TCASH by EtherFlyer.com

Cryptos Trading Made Easy

Providing One-stop Liquidity and Open-source Decentralized Cryptocurrency Exchange Platform

Whitepaper version 3.1 (Draft)

Nov. 05 , 2017

Table of Contents

1.	Abstract	3
2.	Background.....	3
	2.1 Explosive growth of Cryptocurrency	3
	2.2 Centralized Exchange	4
	2.3 Decentralized Exchange	4
3.	EtherFlyer Exchange Platform.....	4
	3.1 Platform Architecture and Modules.....	5
	3.2 Platform Demo	6
	3.3 Features of EtherFlyer Platform	7
	3.3.1 Decentralized	7
	3.3.2 High Speed Ordering and Matching	7
	3.3.3 Cross-Chain Trading.....	7
	3.3.4 Platform Contributors Continue to Win	8
	3.4 Assets that will be Traded Online.....	8
	3.4.1 ERC-20 Tokens	8
	3.4.2 Bitcoins and other non-ERC-20 tokens	9
	3.4.3 EOS and EOS-based Tokens	9
	3.4.4 Fiat.....	9
	3.4.5 Other Assets.....	10
	3.5 Assets Online Management	10
	3.6 Profit Model.....	10
4.	Development Plan and Timeline.....	11
5.	TCASH ICO.....	12
	5.1 TCASH Token Introduce.....	12
	5.2 TCASH Expected Return.....	12
	5.3 TCASH ICO	13
	5.4 TCASH Allocation	14
6.	Fundation Management	14
7.	Our Team	15
8.	Advisors.....	16
	Annex: API Introduction	17
	Official Links.....	27

1. Abstract

This white paper describes EtherFlyer, a decentralized digital currency trading platform based on Ethereum smart contracts that supports ERC-20 type tokens and other token types. The EtherFlyer Encrypted Currency Trading Platform is a one-stop, global trading platform that runs on multiple interactive environments including Windows, Chrome, IE and Firefox, Android, and iOS, delivering a global, weather-ready mix of tokens fluidity. TCASH, a local token for the EtherFlyer trading platform, is tightly controlled at 88 million. Supporting all Ethereum wallets, TCASH is the platform's revenue distribution token, allowing each platform participant to share the opportunities and fortunes of crypto-economic giving.

2. Background

2.1 Explosive growth of Cryptocurrency

Since Satoshi's operation of the Bitcoin blockchain in January 2009, there has been a gradual realization of a decentralized, point-to-point digital currency that maintains its own without any asset guarantees, central issuers value. As the market value of Bitcoin continues to rise, it has attracted the attention of a large public worldwide. At the same time, other cryptographic digital currencies based on blockchain and other technologies have also developed rapidly, with different functions and digital currency for different application scenarios successively developed.

According coincoincap.com website statistics, as of August 2017, the global market value of encrypted currency has reached 100 billion US dollars; the same time, the global daily trading volume reached an alarming 5 billion US dollars. In addition to bitcoin, cryptocurrencies are emerging with new features and technologies, and their total value and volume account for more than 50% of the global market. Currently, the world's top 10 platform daily transaction volume has reached 100 million US dollars, and still maintain the momentum of rapid development.

Among them, in 2015, Turing completed the Ethereum operation, enabling digital currencies to run smart contract functions simultaneously. A smart contract is more than just a computer program that can be executed automatically: it is itself a system participant. It responds to received messages, it can receive and store values, and can send out messages and values as well. All financial contracts can be written as smart contracts in the form of program code.

It is foreseeable that with the progress of blockchain technology, more and more people will migrate from the legal system based on the financial system to the financial system based on the blockchain.

2.2 Centralized Exchange

Centralized exchange trading patterns can be divided into transaction-dominated exchange floor, and quotations dominate the OTC. Because over-the-counter transactions are not transparent and trades involve many risks and uncontrollable factors, people generally choose to trade on the floor. Unless there are special circumstances, they conduct over-the-counter transactions.

The central venue exchange needs to be responsible for functions such as user account management, KYC, asset reloading, asset custody, transaction matching, asset clearing, asset withdrawals, performance guarantee, and new currency launches. However, at every link there may exist Risks and problems, thus giving users a safety loss. Since the MTGOX bitcoin theft, the issue of exchange security is frequent, in the final analysis due to the untrustworthy centralization endorsement as a third party endorsement, inevitably insider operations, hacking, misappropriation of assets, Many security risks.

2.3 Decentralized Exchange

Compared with the centralized exchange, to the central exchange business is much simpler, the user assets always in the user's own or open source under the control of the smart contract code, the user does not need to provide personal information to the exchange, asset transfer does not require any artificial Audit.

Some decentralized exchanges are decentralized in themselves, but they have a central banker at the capital and tokens chargers, and the acceptors receive funds and token acceptance services by charging a fee The bustlingness of these exchanges, the depth of trade is not enough, and many tokens can not buy the dilemma.

Ethereum market capitalization has reached 32 billion US dollars, in line with ERC-20 agreement to develop thousands of tokens, Ethereum-based decenter of the exchange EtherFlyer can be used directly Ethereum wallet address transactions, digital assets are free to enter and exit, not Need to charge the acceptor, the exchange can be short-term high popularity and depth of the transaction.

3. EtherFlyer Exchange Platform

EtherFlyer Exchange Server is a trading backend with high-speed performance, designed for cryptocurrency exchanges. It can support up to 10000 trades every second and real-time user/market data notification though websocket.

3.1 Platform Architecture and Modules

We are building a safety, riskless and user friendly interface for cryptocurrency trading, which we believe EtherFlyer outstands other exchange platforms.

<Architecture>

- Apache Kafka:

A distributed streaming platform. The project aims to provide a unified, high-throughput, low-latency platform for handling real-time data feeds. Its storage layer is essentially a "massively scalable pub/sub message queue architected as a distributed transaction log.

- Redis:

An open source (BSD licensed), in-memory data structure store, used as a database, cache and message broker. In order to achieve its outstanding performance, Redis works with an in-memory dataset.

- MySQL:

An open-source relational database management system (RDBMS). For saving operation log, user balance history, order history and trade history.

<Modules>

- Match Server:

It records user balance and order in memory database. It also writes user history into MySQL, push balance, orders and deals message to kafka.

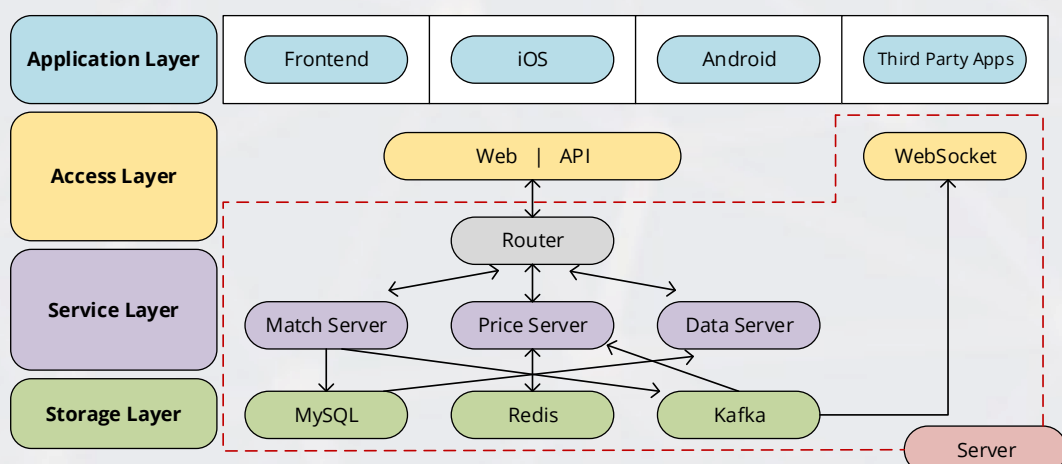
- Market Server:

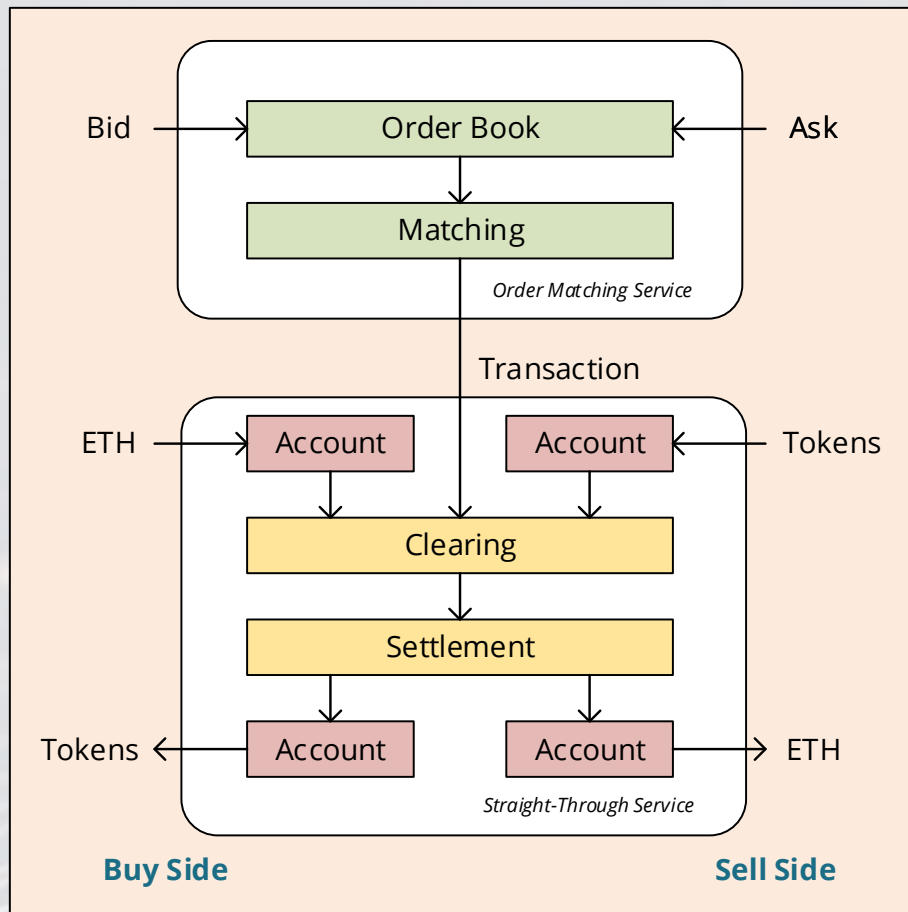
Reads message from kafka, and generates k line data.

- Data Server:

Recover history data from MySQL.

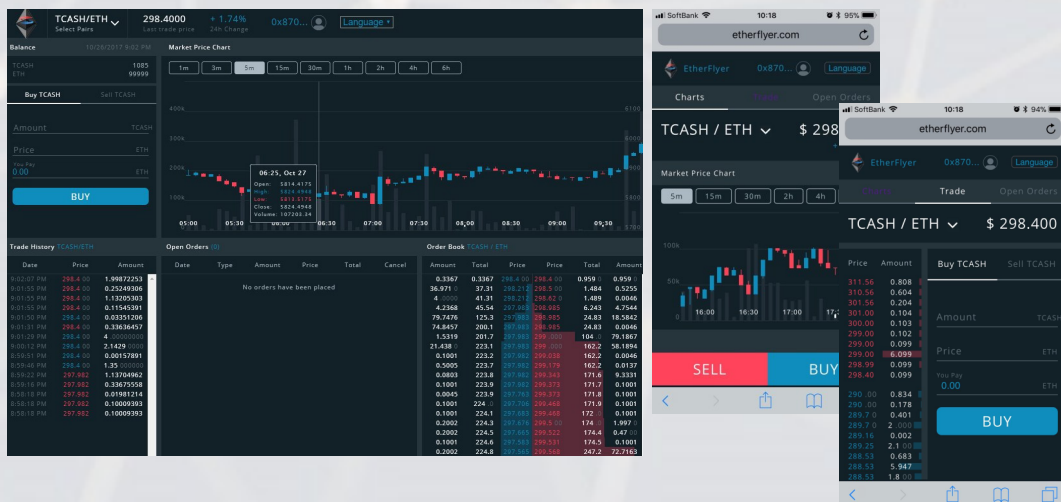
- Web | API :





3.2 Platform Demo

Demo of the EtherFlyer platform is built and testing now which compatible with IE, Chrome, Firefox and mobile phones. The exchange platform is expected to be online in early 2018, stay tuned!



3.3 Features of EtherFlyer Platform

3.3.1 Decentralized

EtherFlyer consists of many contracts running on Ethereum. These contracts interact with each other to allow users to make trades. In order to use the exchange, users enter trade information and send funds to a master contract. Once this master contract receives both sides of the trade, it executes and sends the funds to the respective parties involved. A key distinction between EtherFlyer decentralized exchange and traditional centralized exchanges is where the funds are held during a trade. In the latter case, users are forced to trust the exchange's owners to execute the trade in a fair and secure fashion. In the case of EtherFlyer, the contract - not the exchange - holds the funds. This contract takes the form of auditable code which lives on the Ethereum network. Traders will no longer have to trust private servers to hold the order book, along with every single one of its inputs and outputs. This new order book and the deterministic nature of contracts on Ethereum brings much-needed transparency to cryptocurrency exchanges.

3.3.2 High Speed Ordering and Matching

EtherFlyer is built by means of a number of technologies based on the latest software development requirements. With a combination of the best frameworks at hand, EtherFlyer is aligned with the most recent technology standards. The trading system has adopted the hardware-accelerated and highly-efficient memory matching technology whose order-making peak value is 2 million+/second, order-processing peak value 1 million+/second. These statistics are hundreds times, even thousands of times of the normal industry standard. Therefore it can guarantee no delay or lag throughout the whole core order process when the market booms.

3.3.3 Cross-Chain Trading

EtherFlyer platform for ERC-20-compliant tokens can be directly graded online, and for non-ERC-20 compliant tokens, such as bitcoin, Litecoin, Zcash, etc., the platform will continue to follow the development and gradually adopt different types of digital currency for cross-chain transactions . With the development of Ethereum, Polkadot and Cosmos, the Bitcoin deployment of Rootstock, and the relay technologies developed by other teams, such as BTCRelay, ZECRelay, X-Relay and others, EtherFlyer will use smart contracts to enable cross-link transactions.

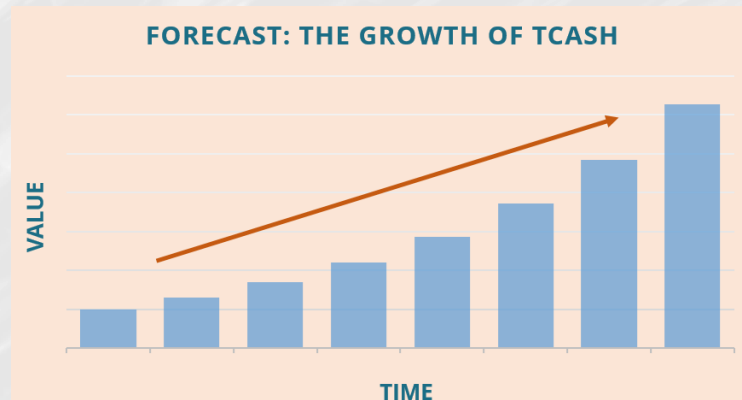
3.3.4 Platform Contributors Continue to Win

All users holding TCASH will share 50% of the transaction fee on the platform. The handling fee will be paid in the form of ETH. The dividend will be distributed to a unified capital pool on a weekly basis. Users can withdraw cash to the user-specified address .

EtherFlyer also reward a discount on the trading fee to user who hold a certain amount of the TCASH token, the more holding amount results in higher trading fee discount. The amount of TCASH holding for trading fee discount are scheduled as:

TCASH Holding Amounts	Trading Fee
Standard Trading Fee	0.30%
200,001~600,000	0.20%
600,001~1,000,000	0.10%
1,000,000+	FREE

























These overall schemes would improve user involvement and exchange vitality. The demand of the TCASH token would also carry out by the definite growth in terms of number of users and trading volume on the EtherFlyer exchange platform. The more in-demand and the higher the token liquidity will be. The value of the TCASH token will definitely be in very high position in long-term and near future.



3.4 Assets that will be Traded Online










3.4.1 ERC-20 Tokens

Any token that conforms to the ERC-20 can be operated online by anyone directly on the web. Officials at EtherFlyer also submit online transactions for mainstream tokens with transaction value every month. When the platform is open for trading, we will first launch the following trading varieties:

 Populous	 OmiseGO	 EOS	 TenX
 0x	 Modum	 Lampix	 Walton
 Bytom	 Loopring	 Status	 Civic
 Monaco	 Storj	 Aragon	 Gnosis
 FirstBlood	 Bancor	 Veritaseum	 Ionomi
 Aeron	 Raiden	 Kyber	

3.4.2 Bitcoins and other non-ERC-20 tokens

The EtherFlyer platform will be systematically developed to expand cross-chain transactions with other major non-ERC-20 virtual currencies, see official announcement for the exact timeline.

 Bitcoin	 Bitcoin Cash	 Litecoin	 NEO
 Zcash	 Tether	 MonaCoin	 Dogecoin
 Lisk		

3.4.3 EOS and EOS-based Tokens

In 2018, EOS platform comes online, any tokens running on EOS platform will be support to trade in EtherFlyer.

3.4.4 Fiat

Although most countries have not yet put in place laws on digital currency and legal currency transactions, in order to avoid unnecessary legal risks, the EtherFlyer platform does not currently open the trading pairs of legal and digital currencies of all countries. In the future, if countries issue relevant legal and legal currency Direct trading, EtherFlyer platform will also open related functions.

3.4.5 Other Assets

The EtherFlyer platform will also develop, in a later phase, the relevant transaction pairs for various assets such as futures, spot and the like, requiring a real-world acceptable acceptor who may be represented by any individual or institution that makes an application and satisfies the relevant requirements.

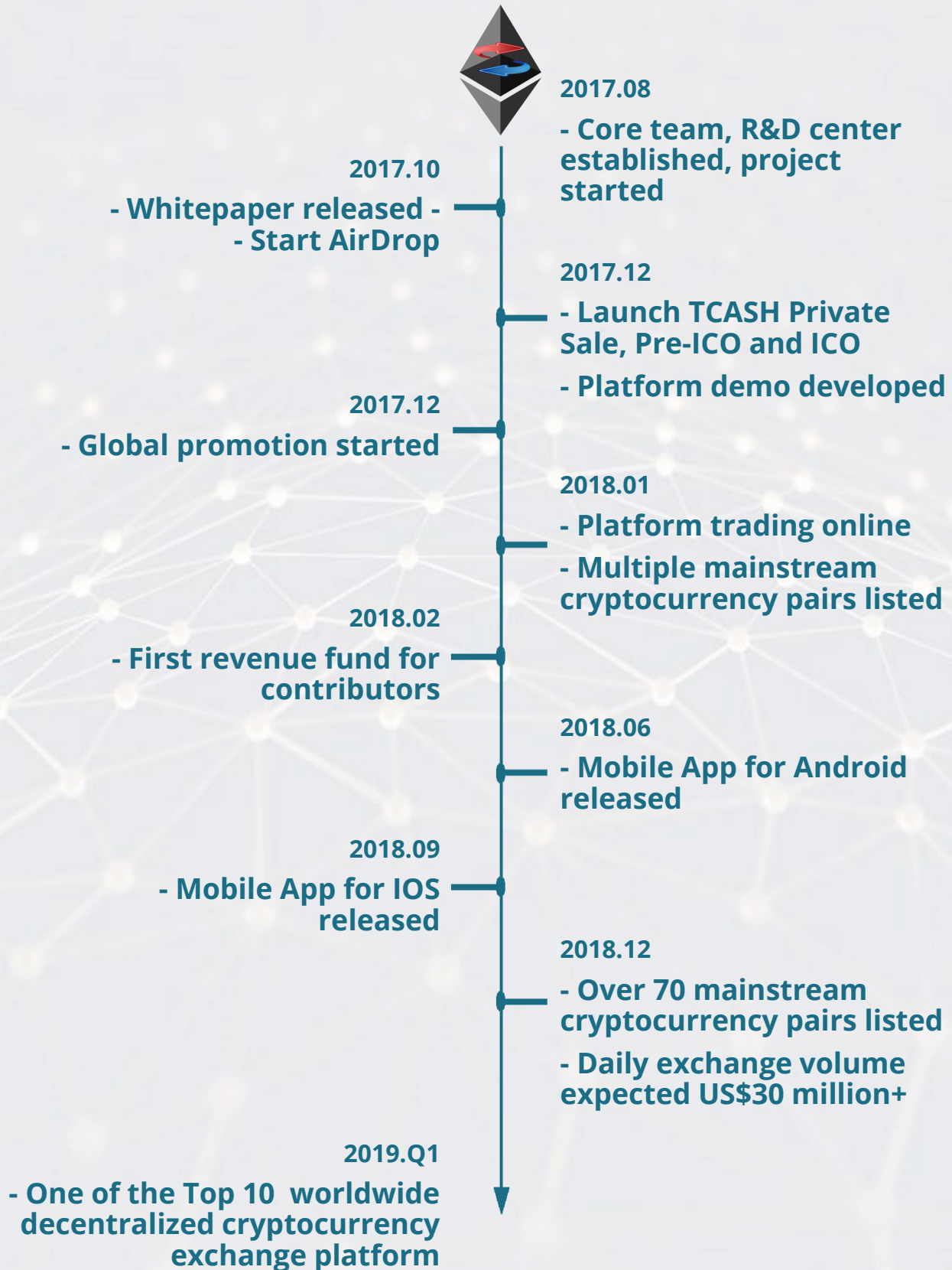
3.5 Assets Online Management

EtherFlyer ERC-20 symbol corresponding to the symbol, free online trading. For non-symbol ERC-20 tokens, can be put on-line application, wait for technical personnel to solve the technical problems, you can online transactions. Some centralized exchanges are linked to the interests of insiders as a result of the assets being shelved. As a result, the exchange only holds trading currencies that maximize its own profits. Many digital assets with real value and technological innovation can not be traded online. Compared to these exchanges, EtherFlyer asset transactions more convenient, only the technical conditions permit, will not cause the user's asset safety risk, that is, online trading. EtherFlyer assets are publicly available online and are subject to oversight by all TCASH stakeholders.

3.6 Profit Model

The TCASH token is a revenue-generating cryptocurrency, that enables TCASH holders to receive revenues sharing from 50% of the total transaction fees charged by the EtherFlyer exchange platform for cryptocurrency transactions. Please note that this revenue sharing proportion could be adjusted based on the actual situation in the future. The revenues are transferred to the TCASH token holders' wallets on a monthly basis. The interest is generated by the transaction fee when transactions in cryptocurrencies supported by EtherFlyer are carried out on the exchange. Holding more than 1(one) TCASH are eligible for the revenues split and must register Ethereum address with EtherFlyer for distribution.

4. Development Plan and Timeline



5. TCASH ICO

5.1 TCASH Token Introduce

EtherFlyer.com is a 100% self-funded project that did not rely on any external investments prior to and upon its launch. The new and very expansive curve of its development implies launching a token native to the platform that will provide contribution opportunities to people who share the EtherFlyer.com vision and believe in the project's potential. The name of the token is TCASH, which is a standard ERC-20 crypto token with total issuance volume of 88,000,000 only and supports all Ethereum wallets.

5.2 TCASH Expected Return

TCASH Revenues Simulation			2017/11/9	
Conditions				
ETHEREUM PRICE		\$300		
TOTAL ISSUANCE AMOUNT		88,000,000TCASH		
TOKEN ICO PRICE		2600TCASH?1ETH		
HOLDE 300,000 TCASH		300,000*300\$/2600=\$34,600		
FEES		2018	2019	2020
24 HOUR TRADE VOLUME		\$5,000,000	\$10,000,000	\$20,000,000
TRADE FEES		0.30%	0.30%	0.30%
FEES /DAY		\$15,000	\$30,000	\$60,000
FEES /MONTH		\$450,000	\$900,000	\$1,800,000
FEES /YEAR		\$5,400,000	\$10,800,000	\$21,600,000
50% REVENUE FUND/YEAR		\$2,700,000	\$5,400,000	\$10,800,000
REVENUE FUND		2018	2019	2020
300,000 TCASH HOLDER		300,000*2,700,000/88,000,000=\$9,200	300,000*5,400,000/88,000,000=\$18,400	300,000*10,800,000/88,000,000=\$36,800
REVENUE FUND /YEAR		\$9,200/\$34,600=26.5%	18,400/34,600=53%	36,800/34,600=106%
REVENUE FUND TOTAL		26.50%	78.50%	184.50%

Coinmarketcap.com 24 Hour Volume Rankings Top 10 (DATE :2017-11-09)

No.	Exchange	Volume (24h)	
1	Bitfinex	\$ 1,526,196,947	
2	Bithumb	\$ 1,101,190,920	
3	Bittrex	\$ 740,356,751	
4	GDAX	\$ 509,698,670	
5	Poloniex	\$ 459,479,747	
6	HitBTC	\$ 310,482,519	
7	Bitstamp	\$ 289,912,180	
8	Kraken	\$ 271,691,407	
9	Binance	\$ 258,794,530	
10	Coinone	\$ 258,397,360	
43	EtherDelta	\$ 9,202,645	

All materials and contents found in this table are strictly for information purposes only and should not be considered as an offer, or solicitation, to deal in any of the crowdsale policies. Any forecasts or projections or forward looking statements made in this Picture are not necessarily indicative of future or likely performance, future performance of trading platform. These statements are only predictions and actual events or results may differ.

5.3 TCASH ICO

There are three phases for the token sale: Private Sale, Pre-ICO and ICO. The maximum number of token is 52.8 million, which accounted for 60% of the total issuance.

Maximum Offered	Phases	Amount	Exchange	Start	End	Days
52,800,000	Private Sale	10,560,000	1 ETH = 2,600 TCASH	2-Dec	9-Dec	8
	Pre-ICO	15,840,000	1 ETH = 2,300 TCASH	10-Dec	12-Dec	3
	ICO	26,400,000	1 ETH = 2,100 TCASH	13-Dec	18-Dec	6

Private Sale Phase: Starting on December 2, 2017, for 8 days, ending on December 9, the total amount would be 10,560,000, representing 20% of the total sales tokens, and one ether can be exchanged for 2,600 TCASH.

Pre-ICO Phase: Starting on December 10, 2017, for 3 days and ends on December 12. The total amount of pre-ICO token is 15,840,000, counted for 30% of the total sales tokens and one ether can be exchanged for 2,300 TCASH.

ICO Phase: The beginning of time is on December 13, 2017, for 6 days until December 18, the total amount of tokens for ICO is 26,400,000, which is counted for 50% of the total sales tokens and one ether can convert 2,100 TCASH.

The token sale also support contribution in EOS tokens, specific information will be announced on the official website <https://tcash.io>, and TCASH will be released after ICO ends with no lock up period. TCASH is confirmed to be traded on the EtherFlyer, EtherDelta and Hotbit exchanges, more platforms will be available in the future. Pre-ICO and ICO participants need to complete KYC registration in the official website to ensure that all participants are qualified.

The purpose of the KYC whitelist is to prevent activities such as terrorism financing, money laundering and financial fraud, and our team can better understand the first phase of TCASH tokens and carefully manage the risks. We will use the KYC standards common to global banks and financial institutions for review. The review included screening of international sanctions / terrorist lists, politically sensitive individuals and persons with negative news, risk assessment using proprietary software, and determining participants' inclusion.

5.4 TCASH Allocation

Total amount of TCASH will be 88,000,000 only. The initial issue of TCASH is programmed by a smart contract and is allocated as follows:

TCASH Allocation	100%	88,000,000
Offered for ICO	60%	52,800,000
AirDrop	1%	880,000
Reserved Fund	20%	17,600,000
EtherFlyer Team	19%	16,720,000
Total	100%	88,000,000

The token allocated for the reserved fund and Etherflyer team will be locked up for six months till June 30th 2018, and the holders for the allocation are prohibited to assign or sell their TCASH holdings in any way during the locking period. The funds, collected during ICO will be spent in the following order:

- ◇ 43% - IT Development, User Experience, Roadmap Completion
- ◇ 43% - Operations and Marketing
- ◇ 14% - Auditing Compliance

6. Foundation Management

The EtherFlyer Foundation will be registered in Hong Kong, the international financial center, headed by top executives from top public companies. The mode of organization of the foundation is subject to the supervision and examination of the relevant departments in Hong Kong. Under the supervision, the income and purposes of the project will be strictly restricted and can only be used for further development beyond the daily expenses. The establishment of the Etherflyer Foundation, which is to be legally compliant, will have a positive impact on the added value of TCASH.

The Etherflyer Foundation adopts a new flat centering and decentralized governance structure with members all over the world to ensure continuity of work while maintaining a clear architecture that ensures professional and efficient work patterns and sustainable development. R&D base in Tokyo, the main members from Japan's leading financial IT companies, for 10 years for large securities companies, state-owned, commercial banks and consortia to provide the overall transaction platform solution technology and experience accumulation.

Strategy and Compliance: The compliance department is set up in Tokyo. The team attaches great importance to governance and compliance. With the global economy of blockchains going forward and the regulatory authorities in various countries and regions varying in their attitudes, it is necessary to reduce policy risks and protect investment Property safety. Operating segments at this stage set up in Shanghai, Toronto, London, Xuanfa, customer service-oriented, for the world's largest markets in Asia, North America, Europe to ensure that the project around the clock.

DISCLAIMER: This whitepaper is for discussion purpose only. EtherFlyer does not guarantee the accuracy of the conclusions reached in this whitepaper. TCASH is not intended to constitute a security in any jurisdiction. This whitepaper is not a solicitation for investment and does not pertain in any way to an offering of securities in any jurisdiction. Please note that purchases of TCASH are final and non-refundable.

7. Our Team



Tomi Xue, PhD, CEO and Founder

Experienced entrepreneur with backgrounds in mathematics, computer science and investment economics as well as consulting, IT solutions. He is CEO of Smart Corporation in Japan, provides trading system with high-security and high-performance as well as website for big securities companies, such as Japanese banks and Nikko Securities, a senior master in cryptography and blockchain technology, has a good grasp of the Ethereum smart contract Solidity language, wrote smart contracts of TCASH crowdsale and EtherFlyer exchange.



O. Ho, PhD Economics, M Finance, Chief Strategy Officer

Responsible for budget management, financial modeling, financial reporting and TCASH foundation construction. He is a financial professional with 10+ years of expertise and specialization in quantitative data/statistics analysis, M&A, financial and management reporting and financial analysis with practical experience in top listed companies.



K. Yu, Lead Technology Developer

More than 15 years of experience in the design of Internet solutions and platform architects, skilled in providing solutions for Japanese financial services, he had served as CTO at many Internet companies. Responsible for EtherFlyer platform architecture design and technology upgrades and connection in the future.



J. Wang, Back-end Developer

Japanese system engineer, holds the qualification of securities analyst in Japan, engages in Japan's financial securities market, had design numbers of large securities trading systems, over 17 years of experience in financial securities system development and project management, familiar with the core of trading platform's architecture, skilled in the development. Responsible for the architecture solution of EtherFlyer trading platform ,back-end system design and application services development.



W. J. Ding, Data Analyst

Machine learning and pattern recognition expert, with rich experience in system architecture development, excellent programming skills such as Java, Python, and JS. Engaged in big data mining and analysis, big data project architecture for a long term, full skilled of handle and analyze large amounts of data, can use various types of machine learning algorithms to build prediction models, brilliant in e-commerce recommendation system architecture, mail recommendation and classification system, real-time monitoring system and data visualization system.



O. Zhan, Marketing Coordinator

Blockchain technology enthusiast, confident of the future of this technology, pay close attention to cryptocurrency development and market for a long run, has rich experience in operations management in industrial of Fortune 500 companies, a senior business development manager, and responsible for EtherFlyer promotion and marketing, to help more people know and use EtherFlyer platform, help more digital assets list online.

8. Advisors



Munetoshi Ozawa - Mitsunari Corporation

Early investment and financing consultant in the field of e-commerce, real estate and other fields, rich experience in the financing of financial institutions, has deep insight in current blockchain and cryptocurrency innovation and venture invest, and leading TCASH fund crowdsale's plan.



Yasushi Kameyama - Angel investors

Successful entrepreneur, angel investor, with many years experience of angel invest, built a investment collectibles club, trade with 36 countries and regions worldwide, will give TCASH Fund early investment advices and global marketing guidance.



TangleCapital -- Angel investors

Tangle Capital, founded in 2017, with primary focus on investment in digital asset including but not limited in decentralized ledger projects. The founding partner of Tangle Capital has extensive experience investing in cryptocurrency since 2011, was early investor of Bitcoin, Nxt, IOTA and Binance.

Annex: API Introduction

The EtherFlyer APIs are designed to allow access to the features of the EtherFlyer platform. The end goal is to allow people to potentially recreate the entire platform on their own. APIs are separated into two categories: trading and Market. Trading APIs require authentication and provide access to placing orders and other account information. Market APIs provide market data and are public. Please note the exact logic and API may change over time to further improve reliability.

API Invocation Instruction

1. All requests based on the Https protocol should set the request header information Content-Type as:'application/json'.
2. All input and output data are in JSON format.
3. Visit Root URL : <https://api.etherflyer.com>
4. Signature is required for trading API related interfaces. The signature data is placed in the authorization header of the HTTP header, and authorization is the signature result string. No signature is required for market API related interfaces.

● **Market Data**

- ◇ Request description: Return the ticker for a given market
- ◇ Request type: GET
- ◇ Signature required: No
- ◇ Call:
https://api.etherflyer.com/market_ticker?market=TCASHETH
- ◇ Request parameter:

name	type	required	description
market	String	Yes	market

- ◇ Output

name	type	description
date	Interger	server time
ask	String	ask price
bid	String	bid price
high	String	24H highest price
low	String	24H lowest price
vol	String	24H volumn
last	String	latest price

Sample:

```
{
  "code": 0,
  "data": {
    "date": 1405699200,
    "ask": "1.02589999",
    "bid": "1.0251",
    "high": "1.0256",
    "low": "1.0249",
    "vol": "245.82513926",
    "last": "1.0251"
  },
  "message": "Ok"
}
```

● **Order Book**

- ◇ Request description: Returns the order book for a given market
- ◇ Request type: GET
- ◇ Signature required: No
- ◇ Call:
https://api.etherflyer.com/order_book?market=TCASHETH&limit=1
- ◇ Request parameter:

name	type	required	description
market	String	Yes	market
limit	Interger	No(Default100)	Return amount: 1/5/10/20/30/50/100

- ◇ Output:

name	type	description
asks	Array	Seller depth
asks[0][0]	String	Order price
asks[0][1]	String	Order amount
bids	Array	Buyer depth
bids[0][0]	String	Order price
bids[0][1]	String	Order amount

Sample:

```
{
  "code": 0,
  "data": {
    "asks": [    # Sell Depth
      [
        "1.021", # Order price
        "500.00" # Order amount
      ]
    ],
    "bids": [    # Buy Depth
      [
        "1.02", # Order price
        "300.00" # Order amount
      ]
    ]
  },
  "message": "Ok"
}
```

● **Trade History**

- ◇ Request description: Returns latest transaction data for a given market
- ◇ Request type: GET
- ◇ Signature required: No
- ◇ Call:
https://api.etherflyer.com/trade_history?market=TCASHETH&limit=1
- ◇ Request parameter:

name	type	required	description
market	String	Yes	market
limit	Integer	No(Default100)	Return amount: 1/5/10/20/30/50/100

- ◇ Output:

name	type	description
date	Integer	Transaction time
type	String	Buy, Sell
amount	String	Transaction amount
price	String	Transaction price
total	String	Transaction total price

Sample:

```
{
  "code": 0,
  "data": [
    {
      "date": 1405699200,
      "type": "buy",
      "amount": "250",
      "price": "1.0259",
      "total": "256.475"
    }
  ],
  "message": "Ok"
}
```

● **Chart Data**

- ◇ Request description: Returns candlestick chart data for a given market
- ◇ Request type: GET
- ◇ Signature required: No
- ◇ Call:
https://api.etherflyer.com/chart_data?market=TCASHETH&type= 5min
- ◇ Request parameter:

name	type	required	description
market	String	Yes	market
type	Interger	Yes	1min,5min,15min,30min,1hour,4hour,12hour,1day,1week

- ◇ Output:

name	type	description
data[0][0]	Interger	Time
data[0][1]	String	Opening price
data[0][2]	String	Closing price
data[0][3]	String	Highest price
data[0][4]	String	Lowest price
data[0][5]	String	Volume

Sample:

```
{
  "code": 0,
  "data": [
    [
      1492358400, # Time
      "1.0204545", # Opening
      "1.0227592", # Closing
    ]
  ]
}
```



```

        "1.025388", # Highest
        "1.0203001", # Lowest
        "44.11655644" # Volume
    ],
    ...
],
    "message": "Ok"
}

```

● **Limit Order**

- ◇ Request description: Places a limit order in a given market
- ◇ Request type: POST
- ◇ Signature required: Yes
- ◇ Call:
https://api.etherflyer.com/limit_order
- ◇ Request parameter:

name	type	required	description
access_id	String	Yes	access_id
market	String	Yes	market
type	String	Yes	sell: sell order; buy: buy order;
amount	String	Yes	order amount, min 0.01
price	String	Yes	order price

Sample:

```

{
    "access_id": "BFFA64957AA240F6BBEA26F4E07EC0D9",
    "market": "TCASHETH",
    "type": "sell",
    "amount": "500",
    "price": "1.025388"
}

```

- ◇ Output:

name	type	description
amount	String	order count
avg_price	String	average price
create_time	Interger	time when placing order
deal_amount	String	count
deal_fee	String	transaction fee
deal_money	String	amount
finished_time	Interger	complete time
id	Interger	Order No.
maker_fee_rate	String	maker fee

market	String	market
order_type	String	limit: limit order; market: market order;
price	String	order price
status	String	not_deal: unexecuted; part_deal: partly executed;
taker_fee_rate	String	taker fee
type	String	sell: sell order; buy: buy order;

Sample:

```
{
  "code": 0,
  "data": {
    "amount": "56.5",
    "avg_price": "11641.3",
    "create_time": 1496798479,
    "deal_amount": "56.5",
    "deal_fee": "1315.4669122",
    "deal_money": "657733.4561",
    "id": 300001,
    "left": "0",
    "maker_fee_rate": "0.001",
    "market": "TCASHETH",
    "order_type": "limit",
    "price": "1.025",
    "status": "done",
    "taker_fee_rate": "0.002",
    "type": "sell"
  },
  "message": "Ok"
}
```

● **Cancel Order**

- ◇ Request description: Cancel unfinished order
- ◇ Request type: DELETE
- ◇ Signature required: Yes
- ◇ Call:
https://api.etherflyer.com/cancel_order
- ◇ Request parameter:

name	type	required	description
access_id	String	Yes	access_id
market	String	Yes	market
order_id	Integer	Yes	Unfinished order No

Sample:

```
{
  "access_id": "BFFA64957AA240F6BBEA26F4E07EC0D9",
  "market": "TCASHETH",
  "order_id": 300001
}
```

◇ Output:

name	type	description
amount	String	order count
avg_price	String	average price
create_time	Integer	time when placing order
deal_amount	String	count
deal_fee	String	transaction fee
deal_money	String	amount
finished_time	Integer	complete time
id	Integer	Order No.
maker_fee_rate	String	maker fee
market	String	market
order_type	String	limit: limit order; market: market order;
price	String	order price
status	String	not_deal: unexecuted; part_deal: partly executed;
taker_fee_rate	String	taker fee
type	String	sell: sell order; buy: buy order;

Sample:

```
{
  "code": 0,
  "data": {
    "amount": "56.5",
    "avg_price": "11641.3",
    "create_time": 1496798479,
    "deal_amount": "56.5",
    "deal_fee": "1315.4669122",
```

```

    "deal_money": "657733.4561",
    "id": 300001,
    "left": "0",
    "maker_fee_rate": "0.001",
    "market": "TCASHETH",
    "order_type": "limit",
    "price": "1.025",
    "status": "done",
    "taker_fee_rate": "0.002",
    "type": "sell"
  },
  "message": "Ok"
}

```

● **Unfinished Order List**

- ◇ Request description: Unfinished Order List
- ◇ Request type: GET
- ◇ Signature required: Yes
- ◇ Call:
https://api.etherflyer.com/order_pending
- ◇ Request parameter:

name	type	required	description
access_id	String	Yes	access_id
market	String	Yes	market
limit	Integer	Yes	Return amount

Sample:

```

{
  "access_id": "BFFA64957AA240F6BBEA26F4E07EC0D9",
  "market": "TCASHETH",
  "limit": 50
}

```

- ◇ Output:

name	type	description
amount	String	order count
avg_price	String	average price
create_time	Integer	time when placing order
deal_amount	String	count
deal_fee	String	transaction fee
deal_money	String	amount
finished_time	Integer	complete time
id	Integer	Order No.
maker_fee_rate	String	maker fee

market	String	market
order_type	String	limit:limit order; market:market order;
price	String	order price
status	String	not_deal: unexecuted; part_deal: partly executed;
taker_fee_rate	String	taker fee
type	String	sell: sell order; buy: buy order;

Sample:

```
{
  "code": 0,
  "data": [
    {
      "amount": "56.5",
      "avg_price": "11641.3",
      "create_time": 1496798479,
      "deal_amount": "56.5",
      "deal_fee": "1315.4669122",
      "deal_money": "657733.4561",
      "id": 300001,
      "left": "0",
      "maker_fee_rate": "0.001",
      "market": "TCASHETH",
      "order_type": "limit",
      "price": "1.025",
      "status": "done",
      "taker_fee_rate": "0.002",
      "type": "sell"
    }
  ],
  "message": "Ok"
}
```

● **My Trade History**

- ◇ Request description: Returns your trade history for a given market
- ◇ Request type: GET
- ◇ Signature required: Yes
- ◇ Call:
https://api.etherflyer.com/my_trade
- ◇ Request parameter:

name	type	required	description
access_id	String	Yes	access_id
market	String	Yes	market
limit	Integer	Yes	Return amount

Sample:

```
{
  "access_id":" BFFA64957AA240F6BBEA26F4E07EC0D9",
  "market":"TCASHETH",
  "limit": 50
}
```

◇ Output:

name	type	description
create_time	Interger	time when placing order
amount	String	transaction amount
price	String	transaction price
fee	String	transaction fee
total	String	transaction total price
type	String	sell: sell order; buy: buy order;

Sample:

```
{
  "code": 0,
  "data": [
    {
      "create_time": 1496798479,
      "amount": "56.5",
      "price": "1.025",
      "fee": "0.047",
      "total": "57.9595",
      "type": "sell"
    }
  ],
  "message": "Ok"
}
```

TCASH Website

<https://tcash.io>

EtherFlyer Cryptocurrency Exchange Platform

<http://etherflyer.com>

Twitter

<https://twitter.com/etherflyercom>

Telegram

[English Group] <https://t.me/etherFlyer>

DISCLAIMER:

You are not eligible to and you shall not purchase TCASH through its token sale if you are a citizen or resident (tax or otherwise) of any country or state where the purchase of TCASH or similar Crypto Token may be prohibited or the token sale is deemed to be not compliant with the applicable laws and regulations. Individuals, businesses, and other organizations should carefully weigh the risks, costs, and benefits of acquiring TCASH. Purchases of TCASH should be undertaken only by individuals, entities, or companies that have significant experience with, and understanding of, the usage and intricacies of Crypto Tokens and blockchain based software systems. Purchasers should have functional understanding of storage and transmission mechanisms associated with other Crypto Tokens. Individuals, businesses, and other organizations should carefully weigh the risks, costs, and benefits of acquiring TCASH.