ParkApp Pay

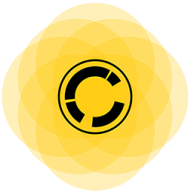
CAR token | Proof-of-Payment Protocol

Automatic IoT Platform

DECENTRALIZING PAYMENTS
YOU MAKE BEHIND THE WHEEL

SUPPORTED BY

Ford



ParkApp Pay

CAR token | Proof-of-Payment Protocol

Technical Whitepaper

Updated at March 1, 2018

CAR token proof-of-Payment protocol (PoP) is a standard that enables new generation of payments the driver could make, being the wheel. We are building Internet of Things (IoT) on the principles of Payment-as- a-Service (PaaS), powered by Internet of Things (IoT) and Artificial Intelligence (AI) technologies and existing IoT protocols like MQTT and protocol buffers.

CONTENTS

1 Introduction

1.1 Disclaimer

You are reading The Technical White Paper.
It was developed by the CAR token team and describes a decentralized technology of payments based on the Proof-of-Payments Protocol.

We provide the description of the technology, based on our level of knowledge and development. We hope you will find it valuable. However, there are certain commitments we are unable to make in regards to the technology of the protocol. Neither CAR token, nor its suppliers and distributors provide any guarantees regarding the Proof-of-Payment Protocol software, aside from those mentioned in the provided and additional Terms of Use. We take no responsibility in regards to the contents of the protocol software, its special functional capabilities, availability and compliance with your requirements. All services are provided "as is".

Local law of certain countries ensures guarantees such as vendibility, serviceability in certain fields, investor protection and intellectual property rights protection. Aside from situations outlined in the legal system, we exclude any and all implied warranties.

After reading The Technical White Paper you may decide to take part in the development of new decentralized technologies, using your knowledge, time and financial resources. Therefore by reading this text, you assume the unconditional obligation that, in the event of being a citizen of USA, China, Singapore, Russia or any other country, any lawsuit with any claimant, where your name is featured as an involved party, we receive a guaranteed right to charge you as a private party for the full amount of losses, including any fines or legal costs, including in the event of your using software (VPN, Class Action, etc) to conceal your true country of residence.

Participants of the token sale accept and agree with following:

- Terms placed at the following URL are final but could be changed by the Company ParkApp: https://parkapp.io/useterms.pdf and https://parkapp.io/agreement.pdf ;
- We will accept ETH, BTC cryptocurrency and fiat currencies for CAR token;
- No returns and no chargebacks are accepted.

ParkApp is not liable for:

- ETH, BTC and fiat currencies rate fluctuations;
- Buyers (anyone who purchases CAR token) actions which contradict instructions or agreement provided by ParkApp, including wrong address/account money transfer;
- Loss of access to a user's crypto-wallet.

1.2 Proof-of-Payment Protocol

Description of the proof-of-payment protocol

The PoP protocol works on top of the protocol buffers https://developers.google.com/protocol-buffers/docs/overview. All message objects generated by the Proof-of-Payment protocol are transmitted using Protocol Buffers.

The basic stages of the Proof-of-Payment protocol
* "Handshake"
* Payment data transfer
* Acceptance of payment reply

General format of message structure

```
message_type {
        optional block_id;
        optional client_token;
        required access_token;
        required action;
        optional data;
        optional version;
        required hash;
        optional error_code;
        optional error_message;
}
```

message_type - message type
block_id - block id in the block where this message is fixed. Used only for the most important messages.
identificator - the identifier of the client
authorization - access token obtained when authorizing a client
action - action identifier for execution
data - the data block required to perform the action.
version - protocol version
hash - hash of all parameters sent in the message for authentication and data integrity

Description of the protocol

Pre-authorization
Before you can start using POP, the client must be authenticated on the server and get access_token.

Handshake Step
After the authorization, the client must pass the "handshake" handshake.
To do this, the client generates a Handshake message:

Handshake {
client_token; - device identifier used when accessing access_token;
access_token;
action: "client_handshake";
version: "1.0" - POP protocol version identifier
hash;
}

In the case of a positive decision, a response from the server should come from him:
Handshake {
action: "success"
block_id - the identifier in the block in the unit where the record about connecting the client to the payment protocol Proof-of-Payment (POP) is stored
hash;
}

in the case of an access_token, client_token, version or hash error:

Handshake {
action: "failed"
error_code;
error_message;
hash;
}

Carrying out of payment

Payment for the selected service takes place in 4 stages:

1. Obtaining information about the selected service. This stage passes using the standard API and the http / https protocol and therefore is not considered further. It is assumed that all the necessary information for the payment has already been received and an invoice has been issued for the payment of a specific service. (REST API)
2. Sending a payment signal using the Proof-of-Payment protocol with a record in the block.
3. Receiving the result of the registration of payment and passing the initial verification of the transaction. (POP)
4. Obtaining the result of the payment made using POP.

Step of sending a signal about payment
A message is generated about the payment for the selected service in the format:

Payment {
access_token;
action: "parking_pay";
payment_method; - payment method ID
invoice_id; - Invoice identifier for the payment of the selected service.
hash;
}

In case of successful registration of the payment signal, the answer will be received:

Payment {
action: "parking_payment_in_process";
block_id - the identifier in the block in the unit where the record is stored about the received request for payment under the protocol Proof-of-Payment (POP)
hash;
}
in case of error:
Payment {
action: "payment_invoice_failed"
error_code;
error_message;
hash;
}

An error at this stage may occur in the case of:
* invoice invalid_id
* Invalid payment method
* authorization or request validation error

Step of receiving a payment response
Since the payment process can take some significant time, the result of success or error will come with some delay.
If the payment process is successfully completed, you will receive a message:

Payment {
action: "payment_success"
block_id: - identifier in the block in the locker where the record about the made payment by the protocol Proof-of-Payment (POP) is stored
hash;
}

in case of error:

Payment {
action: "payment_failed"
error_code;
error_message;
hash;
}

An error at this stage may occur in the case of:
* Invalid payment method
* Insufficient funds

Payments in CAR token system
Entities

Proof of Payment (PoP) - cryptographic proof of payment made by the user for a service integrated with the CAR token platform

Client - is a freely distributed front-end software, the main purpose of which is making payments using Proof of Payment. This software is primarily focused on iOS and Android platforms

User - an individual end-user of the platform, for the convenience of dealing with the Client, the user is associated with the mobile phone number, bundles of payment cards, vehicle data and payment history, as well as the client (software) versions and, optionally, crypto-currency addresses

Service provider (service owner) - is a partner of a company, which provides services paid for including with the use of PoP.

CAR token platform service - is a data storage and processing service

Acquiring account - an account in the bank for collection of fiat money from users in order to transfer them to Service providers

Payment via blockchain - a platform with an ability to execute smart contracts, on top of which PoP will be implemented. At the moment we consider Ethereum, EOS etc.
The life cycle of payment in the ParkApp Pay system

1 Service identification

The client sends a request to retrieve the service data. This part will not require users' manual participation, in such case, the location of the object that provides service is it's identification in our system. For example, the location of an on-street parking zone, tolls, bridges, will be a service identifier. In the future, with the development of technology and the hardware, RFID, BLE, etc. can be used to pay for services that are impossible or difficult to identify by GPS (underground and multilevel objects). In this case, processing such request will be completely decentralized and assigned to the service providers themselves.

2 Payment Authorization

The client receives data from the CAR token platform or the provider and notifies user about cost of the service to authorize the payment. For user-defined "trusted" providers, such as municipal parking, authorization may not be required, then the entire process is carried out automatically.

3 Service Payment

The customer pays the service with the help of ParkApp's acquiring account. ParkApp service allows payments to its acquiring account without user intervention, if the user has already authorized the credit card. In the future, it will be possible to make payments in crypto-currencies or using CAR tokens, without the participation of traditional banking mechanisms. The user will also be able to entrust crypto-currency funds to CAR token platform for payments without entering a password, without the complexities of migration of keys between smartphones and without waiting for miners payment confirmations.

4 Receiving payment confirmations

The client accepts from the ParkApp service the amount of the commission in CAR tokens and as an acknowledgment of the payment for the service the payment transaction identifier, and also - optionally - this identifier could be signed by the provider of the paid service. If service is paid directly through an acquiring bank or blockchain, the confirmation is received in two stages, with the client using the ParkApp service to notify the supplier of the payment and to receive its signature. At this stage, the user can already use the paid service. At the moment we do not use vendors signatures, but the application of this method in the future will completely eliminate possible disputes between users and suppliers.

5 Proof of Payment Protocol formation and network commission

The client generates a service payment message and sends a signed transaction to the address of the CAR token smart contract, recording the payment data for the block account, which operates the CAR token (at the beginning of the development - Ethereum). See the message format below. In this case, the user actually transfers a predetermined commission in CAR tokens to the account of our platform.

6 Obtaining the PoP transaction ID

The client receives the identifier after the PoP is on a blockchain. At this point, the front-end expects several confirmations of the transaction to be recorded in blockchain, increasing the commission of the miners, if required.

7 Transferring PoP to CAR token platform

The front-end client is given some time to confirm the payment of the platform commission. In order to proceed, client sends the PoP transaction ID to the platform, to be checked if data in the message and the paid commission correspond to the data transmitted by the platform service to the front-end client.

In the case of a prolonged absence of PoP transactions in the blockchain, the user is blocked from using the ParkApp service as an intermediary before publishing the transaction used by the PoP client, and in some cases also returns the main part of the payment and notifies the vendor of the payment problem.

Fields used in Proof of Payment Protocol

required int user_id: the identifier of the user-payer. It does not contain personal data by itself, but they can be issued by the platform upon request from the service provider if disputes occur. To generate user_id, the UUIDv4 algorithm is used.

required int service_provider_id: vendor identifier in the CAR token system
optional int service_id: service identifier in the supplier system. The supplier should be able to calculate the cost of the service from the information encoded by this field

optional word service_data: service data. If having the service_id is not enough to calculate its cost, this field is used, for example, to record the exact start and end time of use, tariff, etc.
int amount: service cost in provider currency, not including platform commission

int commission_amount: the amount of commission in the CAR tokens at the time of payment of the service - the same number of tokens must be transferred from the user's CAR account to the ParkApp account for writing PoP to the block account

int payment_id: transaction identifier in the acquirer bank system or in the payment blockchain
optional word payment_confirmation: payment_id, signed by the service provider's key

1.3 Smart City and IoT Protocols

Smart Parking Specifications

One of the fundamental components of smart parking is the intelligent and rapid payment system. This system is will be actively used in many countries and will be based on contactless methods of payment with the help of CAR token and proof-of-payment protocol.

Auto-payment as a service for automatic payments behind the wheel, will work on a blockchain via Proof-of-Payment protocol, which will be connected directly to the car's on-board to receive data directly from the vehicle. Payments for commercial parking lots, drive-thru fast-food chains, toll roads, bridges will also be made through this protocol to make transaction safer and faster.

Proof-of-Payment is a protocol of payments pre-authorization and execution on which performance of an automatic and contactless payment services is based.
In cooperation with carmakers, the technology of Connected Car and Smart Payment will interact directly with the vehicle's on-board system to completely immerse the user and a car in the ever-increasing concept of Smart City and IoT (no additional equipment required).

Proof-of-Payment is the protocol to confirm the transaction for the service before it could actually be mined or delivered to the service provider (like a parking owner or toll road operator etc). This protocol will allow participants of the system, be it an individual user or a business entity, to rapidly confirm the fact of payment.
ParkApp Pay will develop a system for a quick and reliable exchange of information about the transactions conducted to pay for any services available to the driver. All operations are recorded in the block system using Proof-of-Payment protocol, which provides a reliable and transparent payment service.

ParkApp Pay platform will interact with the vehicle to implement the Proof-of-Payment protocol based on the existing and widely used IOT MQTT protocol.
ParkApp Pay is contracted and will cooperate with OEM's of parking equipment to implement the Proof-of-Payment protocol based on existing solutions NFC, ZigBee, BLE, etc.

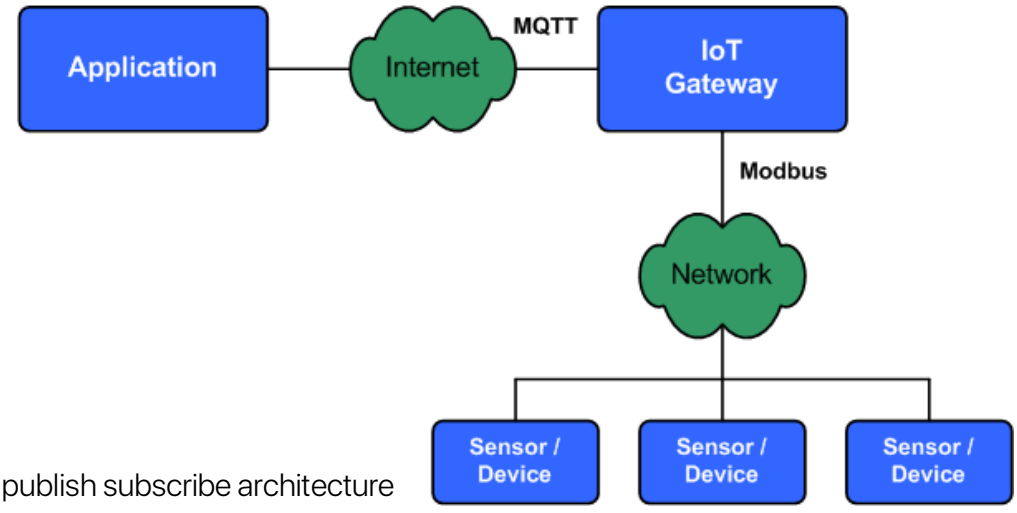
The Proof-of-Payment protocol will be developed to function with existing IoT solutions to ensure a guaranteed instant payment between devices, namely between the vehicle's side and equipment, the vehicle and our platform, the smartphone and the car's side. Also, the purpose of the platform and the protocol is to put transactions on a blockchain, with mandatory encryption of the personal data.

Smart-Car - will allow vehicles to build between each other a smart self-organized cell-network with a relay and routing. The network will duplicate the connection to a more global, but also a distributed network. This will increase the opportunities for interaction between participants over long distances. This technology will allow cars to interact with objects connected to a Smart City to automate most of the routine activities that drivers of a megacity face.

The most common OSI application-level protocols used by devices and apps in the Internet of things are MQTT (remote monitoring of data collected from a large number of devices and their telemetry in the IT infrastructure).

MQTT

MQTT stands for Message QueuingTelemetry Transport. It is a publish/subscribe, extremely simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. The design principles are to minimise network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery. These principles also turn out to make the protocol ideal of the emerging "machine-to-machine" (M2M) or "Internet of Things" world of connected devices, and for mobile applications where bandwidth and battery power are at a premium.



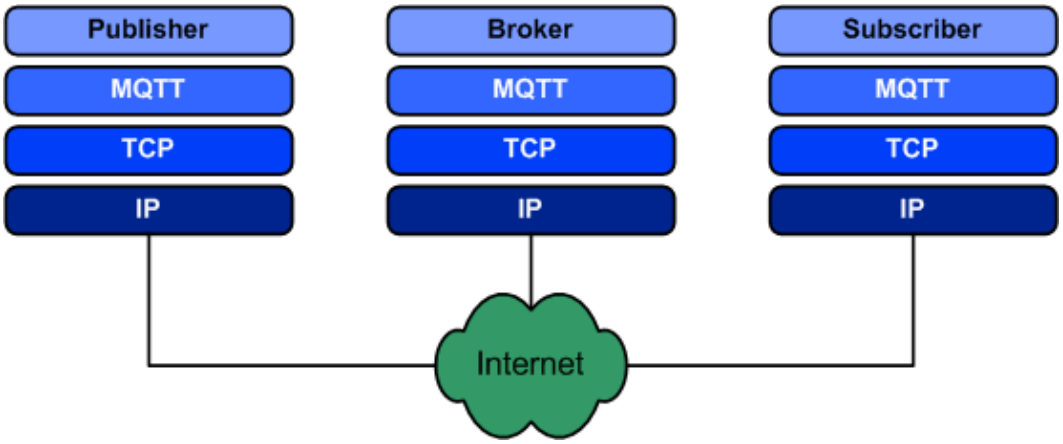
1) MQTT publish subscribe architecture

The MQTT messages are delivered asynchronously ("push") through publish subscribe architecture. The MQTT protocol works by exchanging a series of MQTT control packets in a defined way. Each control packet has a specific purpose and every bit in the packet is carefully crafted to reduce the data

transmitted over the network. A MQTT topology has a MQTT server and a MQTT client. MQTT client and server communicate through different control packets. Table below briefly describes each of these control packets.

2) Ideal for constrained networks (low bandwidth, high latency, data limits, and fragile connections)

MQTT control packet headers are kept as small as possible. Each MQTT control packet consist of three parts, a fixed header, variable header and payload. Each MQTT control packet has a 2 byte Fixed header. Not all the control packet have the variable headers and payload. A variable header contains the packet identifier if used by the control packet. A payload up to 256 MB could be attached in the packets. Having a small header overhead makes this protocol appropriate for IoT by lowering the amount of data transmitted over constrained networks.



3) Quality of Service (QoS) for MQTT

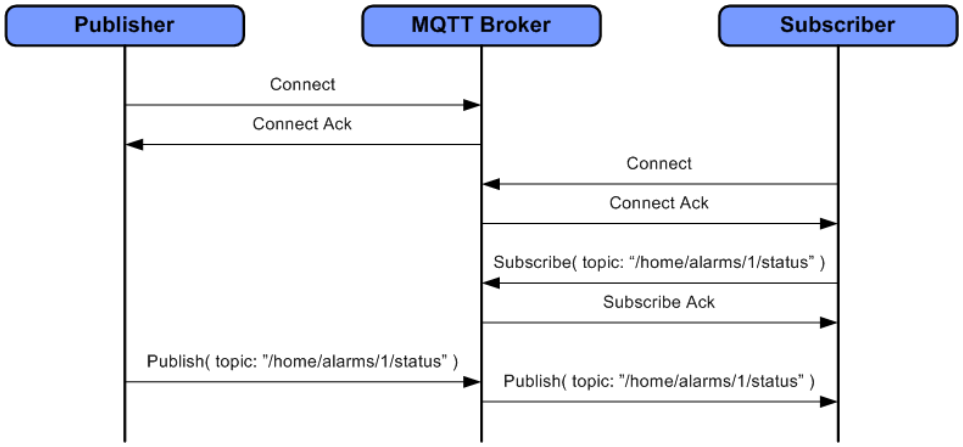
Quality of service (QoS) levels determine how each MQTT message is delivered and must be specified for every message sent through MQTT. It is important to choose the proper QoS value for every message, because this value determines how the client and the server communicate to deliver the message. Three QoS for message delivery could be achieved using MQTT:

- QoS 0 (At most once) - where messages are delivered according to the best efforts of the operating environment. Message loss can occur.
- QoS 1 (At least once) - where messages are assured to arrive but duplicates can occur.
- QoS 2 (Exactly once) - where message are assured to arrive exactly once.

There is a simple rule when considering performance impact of QoS. It is "The higher the QoS, the lower the performance". MQTT provides flexibility to the IoT devices, to choose appropriate QoS they would need for their functional and environment requirements.

4) MQTT client abnormal disconnect notification

When a MQTT client connects to the MQTT server it can define a topic and a message that needs to be published automatically on that topic when it unexpectedly disconnects. This is also called the "Last will and testament" (LWT). When the client unexpectedly disconnects, the keep alive timer at the server side detects that the client has not sent any message or the keep alive PINGREQ. Hence the server immediately publishes the Will message on the Will topic specified by the client. The LWT feature can be useful in some scenarios. For example for a remote MQTT client, this feature can be used to detect when the IoT devices goes out of the network. The LWT feature can be used to create notifications for an application that is monitoring the client activity.



5) MQTT clients are very simple to implement

MQTT is open protocol and standardized by the OASIS Technical Committee. This makes this protocol easy to adopt for the wide variety of IoT devices, platforms, and operating systems. Many applications of MQTT can be developed just by implementing the CONNECT, PUBLISH, SUBSCRIBE, and DISCONNECT control packets. A variety of MQTT client libraries are made available through the Eclipse Paho project. Eclipse Paho MQTT client libraries could be downloaded from the Eclipse Paho website.

The IBM MessageSight messaging appliance helps to deliver the performance, value, and simplicity that organizations need for accommodating this multitude of devices and processing large volumes of events in real-time. IBM MessageSight is built from the ground up with new technology to provide high scalability. IBM MessageSight extends existing messaging networks by adding the following characteristics:

- Fast transaction rates
- Consistent lower latency

- Extensive scaling in the number of concurrent devices that can be connected
- Suitable for deployment in a demilitarized zone (DMZ)

In summary, MQTT is the protocol build for M2M and IoT which can help provide revolutionary new performance and opens up new areas for messaging use cases for billions of things connected through the Internet. The IBM Redbooks publication
Building Realtime Mobile Solutions with MQTT and IBM MessageSight covers the theory and real world scenarios for using MQTT along with IBM MessageSight for mobile and other Internet Of Things applications.

2 Token sale

Total emissions
1 billion of CARTOKEN (CAR).

Token rates:
1 ETH = 8400 CAR tokens
1 BTC = 19600 CAR tokens

Bonuses:
Private Pre-Sale – plus 50% of tokens bought by you.
Main Sale 1st and 2nd week of sale – plus 20% of tokens bought by you then no bonuses till token sale ends

24% of tokens will be reserved for the ParkApp company
8% of tokens will be distributed between ParkApp partners
68% of tokens is allocated for the Main Sale.

50% of funds is allocated for platform, app and company's service marketing and ads
35% of funds is allocated for a research, development, salaries
5% of funds will be spent on OpEx, including the servers rent
10% of funds will be the company's reserve fund

3 INDUSTRY

3.1 Intro

Facts and Some history

Parking space as a market exists since 1898, when the world's first garage was opened on May 24 in Boston, Massachusetts.

Although, there is also a fact about the first multi-level parking, opened in May 1901 in London. So, the need for regulated parking has emerged more than 100 years ago. Since then, the number of cars in the world's big cities has grown dramatically, the number of parking lots, of course, also increased, as well as their cost.

Back in 2009, Boston registered the sale of one parking space to the private owner for $300,000, and today the cost of one hour on commercial parking in Manhattan can reach $60.

Economics

Today, the number of cars in the world reaches a billion, and the number of drivers in the world's megacities is about 700 million people.

The world parking market today reaches more than $ 100 billion dollars of annual turnover and does not stop growing. At the same time, more than 50% of them are accounted for the USA, where ⅓ are the municipal parking lots, and ⅔ commercial ones.

About $ 400 millions annually invested in parking space.

The most profitable parking in the US are the airport lots ($ 2,651 per year from one spot), business centers and corporate parking ($ 2,021 per year from one spot), followed by sport and concert halls, municipalities, hospitals, transit parking and educational institutions.

The demand for parking is not seasonal and there is no dependence on many demographic indicators, such as gender, age and level of education of the driver. Parking is needed in any weather and time of the day, or more precisely, whenever the car is not in motion.

The parking market is growing, firstly, as the increase of number of drivers, cars, roads and the number of citizens coming to work and recreational areas by car. Secondly, as the megacities develop and create the paid parking space. Thirdly, as the cost of parking time itself increases both in municipal and commercial parking as a result of the impact of the first two factors.

The market of parking space at the moment could be described briefly as:

1. The number of privately used cars in the world reaches one billion, and the number of drivers in megacities is about 700 million people, of which more than 50 million are the drivers of US megacities
2. The annual world's turnover exceeds $ 100 billion dollars and rapidly grows every year.
3. More than 50% of the turnover accounted for the US market, ⅓ of which is the municipal parking space, and ⅔ are the commercial one
4. The average driver in the US spends an average $ 1,000 a year on parking

3.2 Problems in the industry

Driver's perspective

1. **Citation**
   In New York, Moscow, Paris and other megacities, the problem of drivers is a municipal parking fine, issued after paying for parking. This is due to a loss of information about transactions for parking payment in the central systems of municipality. Drivers can't appeal against the fine, since the municipality for a range of technical reasons or human factors does not have the transaction data.

2. **Vacant Spot**
   Drivers suffer from searching for a vacant parking spot in big cities. For this problem, there is no technological or peer-to-peer solution. Users spend up to 30 minutes every time they are looking for a spot.

3. **Payments you make being behind the wheel**
   Lack of an automatization of the payments for both cities and commercial parking. No integration with the vehicle on-board system. Often drivers aren't able to pay for a municipal parking due to lack of instructions or it's redundancy, or just being in rush and confused. In such case drivers are being fined or the municipalities collect much less payments.

   There is so called 'sub-problem'. Corporations with the corporate use vehicles park often reimburse employee's transport expenses like gas and parking. But the reimbursement in terms of taxes looks like an extra salary so the employer has to increase the tax payments even more, making this compensations to employees.

4. **Parking Apps**
   Excessive amount of classic, rapidly aging parking apps with narrowed functionality, confusing interface and no wide-range solutions platform behind it. We assume that any driver would express a reluctance of installing more than 1 mobile app for a parking service worldwide.

Problems of municipalities

1. **Centralized data**
   Up to 14% of transactions data is being lost. The problem described above is also a serious and difficult task for the municipalities. Namely, the need to create a service for the decentralized storage of data on transactions that cannot be lost, fabricated or intentionally erased. Even the cashless transaction made on the on-street parking machines are being lost, not verified or under the argument.

2. **Amount of Payments**
   The inconvenience of payment methods motivates the driver to come up with ideas of how to avoid the payment.

3. **Traffic Jams**
   20% of the traffic jams are the slowly moving, looking for parking spot cars, being called for that a "parasite traffic"

Commercial services, parking lots and drive-thru food chains

Loss of customers and time

Drive Thru Chains, Toll Roads, Gas Stations and other facilities, where you have to park your vehicle (with or without the need to leave it) to get the actual service, expressed their concerns to us about the traffic jams. These traffic jams effectively lead to customers dissatisfaction, making them leave the venue, what decreases business activity of such services (except for Toll Roads, of course – you physically can't avoid it if you are there).

On parking lots with manual control of the entrance-exit equipment (barrier) and a small stream of cars, drivers often use the opportunity to negotiate with the attendant to avoid or decrease the payment.

3.3 Existing tools and solutions

1. **On-street parking payments and session extension**
   In some cities there is a phone account payment via SMS. But mobile operators take a high commission and the driver needs to know the code of the parking zone, the service number to send the SMS to, and the text body of the SMS message itself. Mobile applications of municipalities are legislatively limited by the municipal area and cannot proceed payment for non-municipal parking lots.

   The main disadvantage of this approach is that each city has to develop a mobile app to facilitate payment management. This narrows the ability of small cities to integrate such method, and secondly forces traveling drivers to install new apps again and again.

2. **Commercial parking payments**
   Payment is made by credit card or cash at the parking meters, automatic machines or to the attendant.

3. **Searching for an available spot on the on-street parking**
   Drivers create traffic jams, cruising around the final destination searching for a spot or temporarily parking on the road, waiting for a vacant space.

4. **Appealing against unjust fines**
   Long prosecutions and courts with municipal services or the driver's inescapable consent to pay an undeserved fine.

3.4 ParkApp Pay Solution

1. **Blockchain**
   Thanks to our platform and the ParkApp Pay service, all data on payments and transactions will be recorded in the decentralized Blockchain. This will avoid unfairly written fines and further proceedings between the driver and the municipality.

2. **One unified tool**
   We will create a unified app and platform for the entire parking space and payments that the driver can make being behind the wheel. The interface will be automatically adapted for all countries we cover. The driver will no longer be confused by the question: "What app should I download in this city for manual or automatic payments?"

3. **'ParkApp Pay' - Automatic on-street parking payments**
   Our auto-payment system will allow the driver not to worry about the parking zones, rates, payment methods, extension and completion of the parking session. Combining the vehicle's on-board computer with our platform and the mobile app, parking will always be paid correctly, on time and with minimum commission automatically.

   For corporate clients.
   Cars integrated with the 'ParkApp Pay' system will have a great value for those companies who hold their corporate cars for its staff. This system will release the corporations from paying additional taxes of the gasoline and parking expenses reimbursement.

4. **Automatic Payment for commercial parking, food orders in drive-thru chains and toll roads**
   ParkApp will be integrated with banks, parking and processing equipment, allowing you to pay for a service or a food order via ParkApp Pay completely automatically, reducing the car congestion. The license plate will be recognised at the time of entry to the parking lot, passing the toll roads or ordering the food at the drive-thru chain, charging the necessary amount for payment, without drivers' involvement.

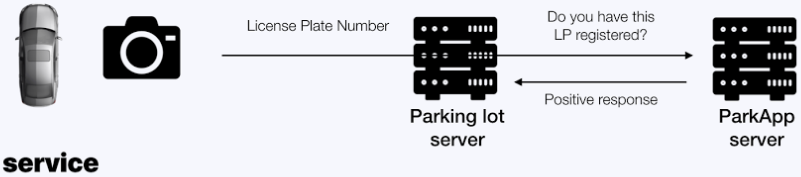
5. **Finding a Vacant Spot**
   The parking module algorithm allows users to see vacant and vacating parking spots in real time, predicting the intention of the driver to leave. Data is being collected from the sensors of the smartphone and on-board vehicle's system (if connected to the ParkApp Pay service). It does not depend on the city infrastructure and calculates congestion forecasts independently.

IMPORTANT: this information is depersonalized and technically optimized for the safety of the user, car and smartphone. Spreading ParkApp among drivers, the traffic congestion will be greatly reduced as well as the time spent by drivers on parking making payments.

ParkApp Pay

Smart ticket less payments

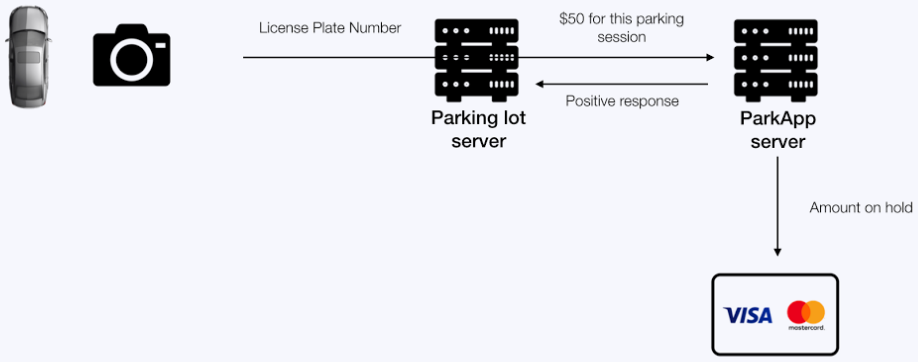
- Vehicle arrives at the entrance of the parking lot.
- Camera reads the license plate at the entrance.
- It matches with one registered in ParkApp.
- The parking session starts and the user drives in.



User is able to partly pay for service with Parkoin tokens

- User's registration in ParkApp.
- Credit card and license plate.
- Crypto wallet settings

- Vehicle arrives at the exit of the parking lot.
- Camera reads the license plate.
- It matches with an open parking session.
- We charge the credit card and open the barrier



Post-pay system

User is able to partly pay for parking with Parkoin tokens

User gets the parking ticket at the entrance. The parking session starts.

1 After some time the user pays for parking

2 Manual input of the ticket number into the ParkApp or the bar-code scanning

Response if such parking session does not exist in case of a incorrect ticket number

3 We send the ticket number to check out the opened parking session

ParkApp server

4 We check out the opened parking session and request the total amount to pay

5 Parking server responds with the amount

Parking server

6 We request the bank to hold the amount needed

7 Then the bank responds that the amount is on hold

10 We confirm that the bank can now charge the user. In case of no response on the stage №10

8 We inform the parking server that the amount is on hold

9 Parking server confirms that the user can leave now and the session is closed

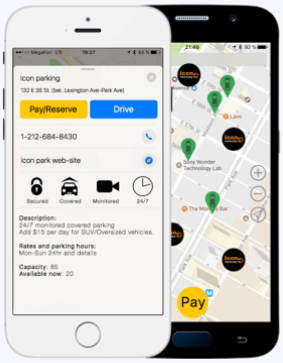
ParkApp server

11 Parking session is closed. We inform user about the time left to leave the parking lot

Pre-pay system

User is able to partly pay for parking with Parkoin tokens

User opens the **ParkApp** and finds the **parking lot's** logo on the map and selects one of the options:
- Pay right now;
- Make a reservation.



We provide the user with rates, details, and the tool to choose the day and time of arrival and departure

Pay now

1

Make a reservation

2
Request:
If there available spot for this period

ParkApp server

Parking lot server

All the parking session are being synchronised with the parking lot server and recorded to Blockchain so the attendant could check the reservation and the payment could not be fabricated or modified.

3
Response:
- Parking is available;
- Please choose another day/time

4

5
Then the bank responds that the amount is on hold

ParkApp server

6
We inform the parking server that the amount is on hold

We request the bank to charge the amount

When the customer arrives, he/she can show the attendant the payment history right in ParkApp with all the details (time, location, amount, licence plate, name)

We detect the server's time-out on every stage. The time-out source is being logged in ParkApp's server and transferred to the parking server also. We inform the user about errors and suggest to pay via parking machine or try again.

4 Platform.Technology

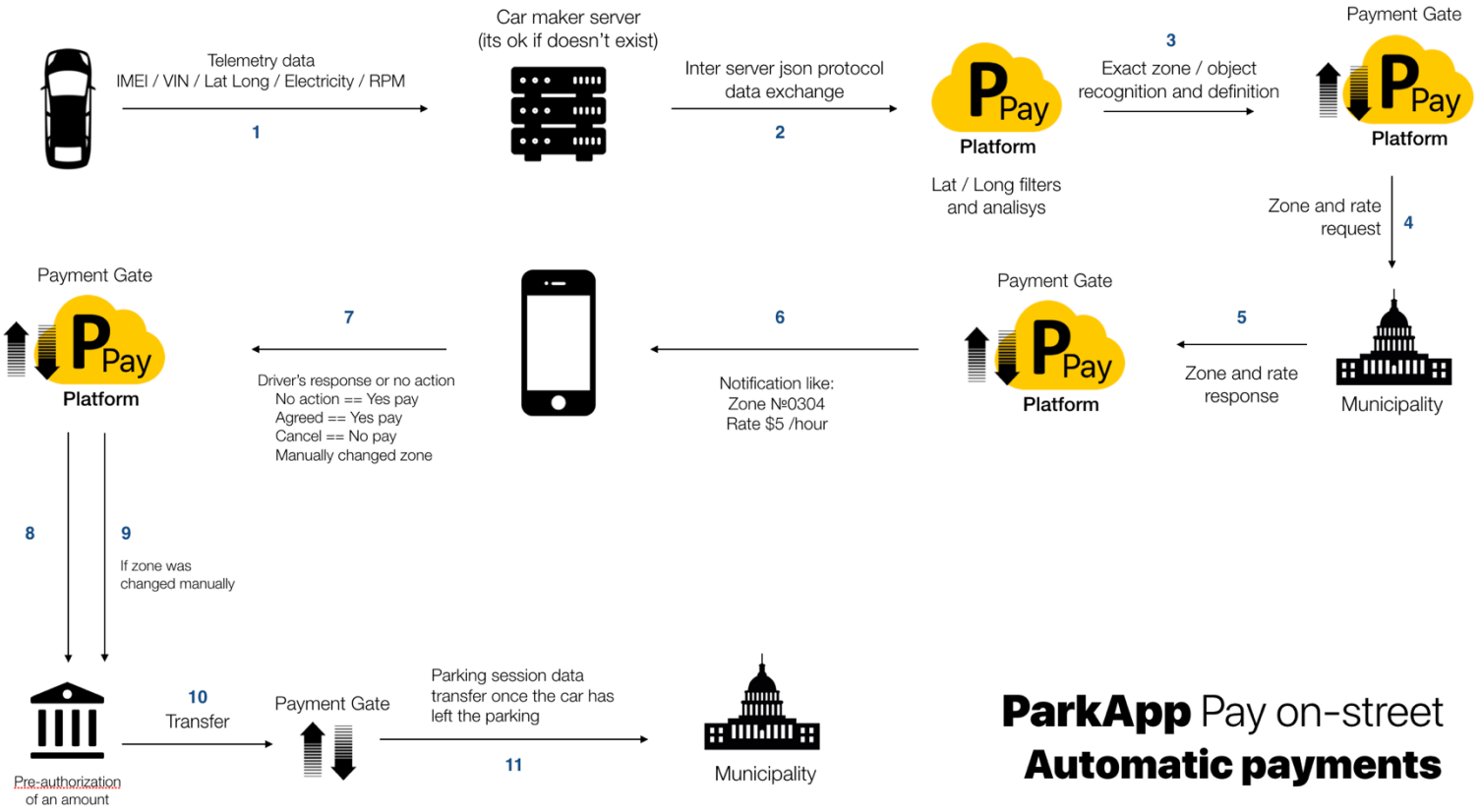
4.1 ParkApp Pay

On-board Automatic Payments

'ParkApp Pay' – automatic payments the driver could make being behind the wheel, without you worrying and participating.

Auto-Payment is applicable both for cars with external trackers, allowing to send coordinates of the stop spot and cars with built-in navigation module. Transmission and analysis of data will be performed in real time with precise detection and distinction between stopping at the traffic lights, in traffic jams or actual parking. Payments will stop automatically upon departure, and the difference between Paid and Used time will be returned back to the user's individual or corporate account (in cases when you prepaid for a full hour, but left earlier).

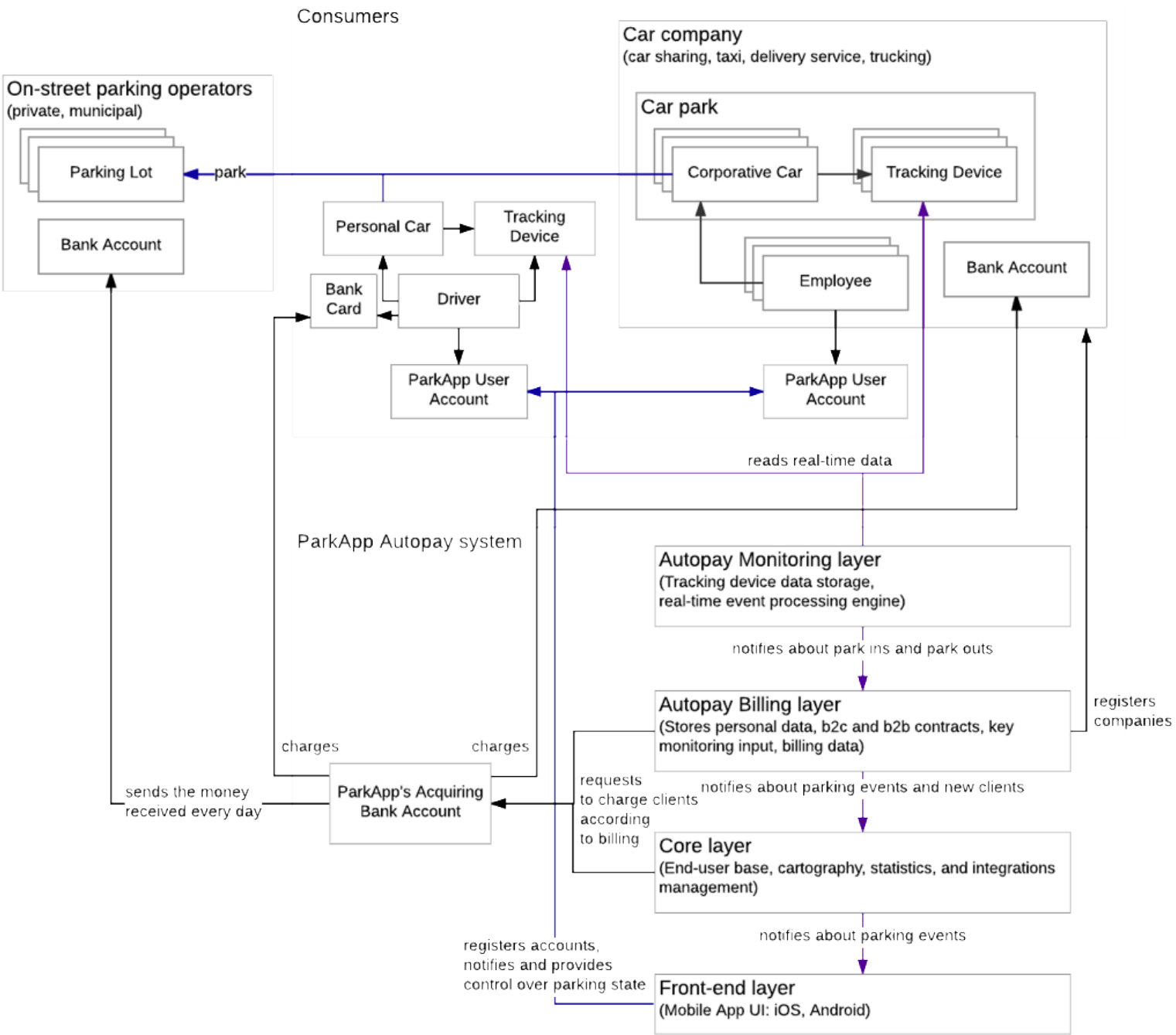
A mobile app is also needed in off-street parking lots for getting notifications or scanning a parking ticket (malls, airports, etc.). The mobile app will show you the parking availability and where you left your car.



What's Good?

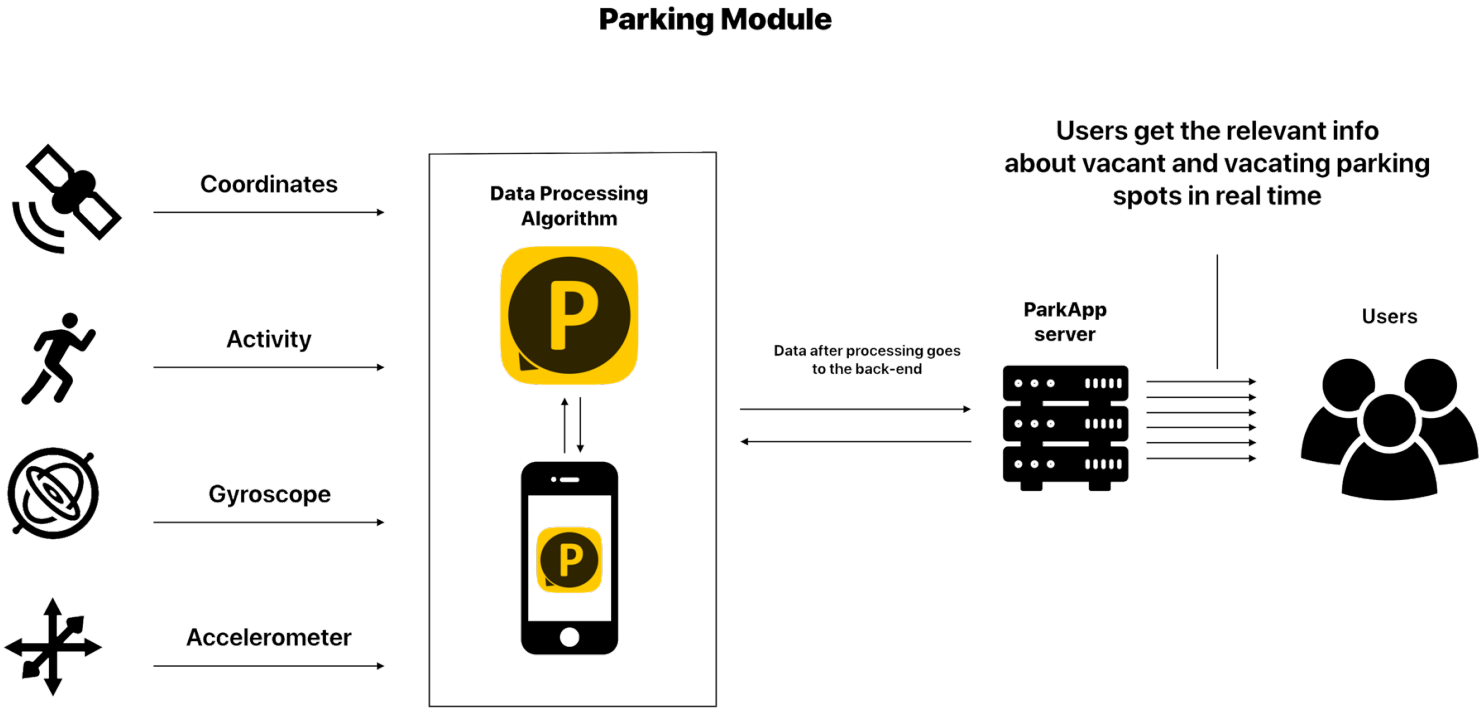
- The additional value and attractiveness of the car brand for corporate clients is much higher if the manufacturer is connected to the 'ParkApp Pay' platform;
- Auto-payment will allow corporate clients to reimburse employee's expenses without paying additional taxes.
- Our product is tested by the Ford Motor Company on a municipal parking space.
- A unified solution for both municipal and private parking in the world.
- Constant support of product stability, regular updates and service improvements.

Autopay scheme



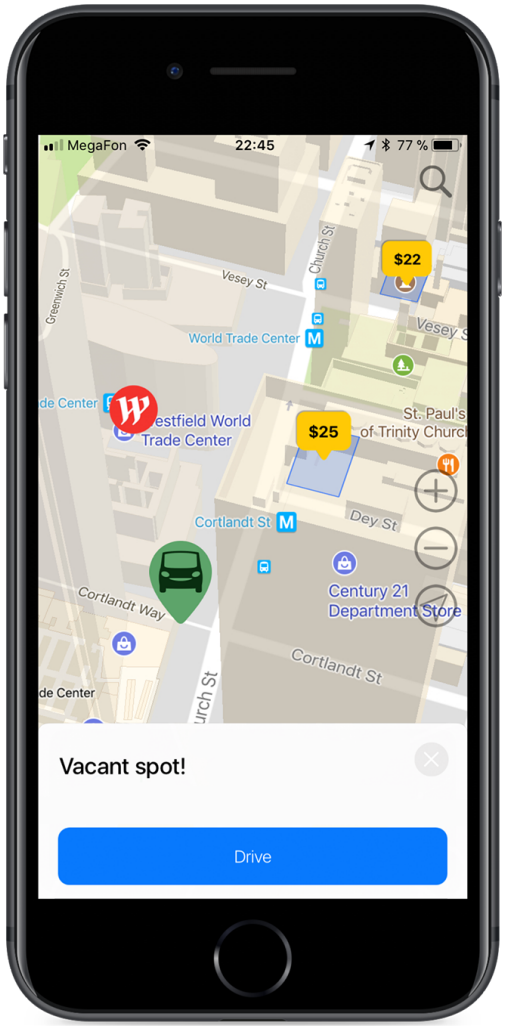
4.2 Forecasting Free Parking Spaces Technology

Parking Module is one of the core ParkApp functionalities. With this module, users get accurate and precise information on available parking spaces in real time, and predictions as to when spaces will become available.



The second important function of the parking module helps you find your parked vehicle. Initially, the program used the GPS sensor only. It is effective but energy-consuming solution. An accelerometer, gyroscope, and Activity Recognition module were added later and, after rigorous testing, became the main sensors in the system.

At the next stage of development, the parking module will use the recurrent neural network, which will do the main work on forecasting available parking spaces by processing temporary data rows. This approach will help make the data on vacant and soon-to-be vacant parking spaces more precise for each ParkApp user.



5 CAR token (CAR)

CAR token is designed to be used within the ParkApp Pay platform and ecosystem.

With the use of an Ethereum smart contract 1 Billion ERC20-compliant CAR token's (CAR) are issued.

No new tokens will be issued.

Unsold tokens will be "burned", as well as 30% of used tokens in the ecosystem will also be burned.

Until the end of the Token Sale we aim to develop the new module Beta version (MVP) of the existing mobile app, which supports the CAR transactions, also will provide the decentralized (peer-to-peer) usage of CAR. Potential advertisers will also be able to place their ads in ParkApp, using CAR token.

CAR token will not be distributed between buyers until we deliver MPV, in order not to speculate.

5.1 Tokenomics

As a result of integration of a key product - ParkApp Pay, users will be able to use CAR within the system as a commission fee for using the service we will provide for drivers.

A user can pay the transaction fee with fiat money as well. However, by using CAR users get an access to all the application features, like partner programs participation, and also protect themselves from unjust fines and citation by putting the data on a blockchain.

At the same time, for a user to buy CAR token for fiat money using the mobile app will be as easy and quick as to use the traditional payment mechanisms.

ParkApp is considering a service transaction fee of 8% as satisfactory for both the company and the US service providers.
Accordingly, the average active user will pay a commission of at least $100 per year = 2000 CAR tokens. This calculation includes parking rent/sell/buy, drive-thru, car insurance, toll roads.

In a pessimistic scenario by the end of 2018, ParkApp unites 1 million active users within US megacities.

Annual usage = 1 million people × $100 = $100 million

In an optimistic scenario by the end of 2019, ParkApp connects 10 million drivers of US megacities and 20 million drivers from other countries.

We will make an assumption that drivers of the US megacities spend on parking more than others, and pessimistically assume that the expenses of the average driver of other countries are four times lower.

What factors might influence the usage of CARTOKEN?

- Cost of ParkApp service fee
- Variety of the platform's functionalities
- The number of active users of the application and the ParkApp platform
- Number of integrated parking lots and carmakers with the app and platform
- The ratio of regulated parking space to the number of cars in the city
- The ratio of the platform users, drivers of megacities to the number of CAR tokens in the system

5.2 Token Usage

How Do I Purchase CAR token?

1. Private pre-sale: is a closed presale among narrowed circle of people. Bonus is 50% sharp for all the buyers. Buyers can use ETH or BTC or any other tokens to buy CAR token.
2. During the main crowd sale period, CAR token can be purchased with ETH or BTC from only one web-site https://parkapp.io . Bonus is 20% first 10 days of sales, then 0%.
2. After the token sale ends. Users could buy tokens from each other, using our platform.

Why is CAR token useful?

1. Transparency and authenticity

- CARTOKEN transactions are transparent because they exist in the blockchain of the decentralized Ethereum network. That means anyone can verify the transfer of CARTOKEN tokens from one user to another.
- CARTOKEN transactions are managed by open-source smart contracts, that is, anyone can verify the logic of transferring tokens between users.
- Transparency allows us to create loyalty programs where end users may receive discounts for the active use of services. The use of those services can be verified by the transaction history offered by relevant third-party services without the need to take the word on trust.

2. Cheap and reliable peer-to-peer (p2p) transactions

- Tokens on the Ethereum network can easily be used for making p2p transactions, that is, exchanging services for "rewards" between end users, with much lower costs than making such transactions through traditional banking systems. The quality of each individual exchange can be assessed by the users themselves by means of reputation points associated with the p2p transaction.
- When making p2p transactions that are paid with fiat money and concluded "at your word", situations may occur when a consumer does not actually have sufficient funds. Blockchain allows you to perform a procedure in the same way as "pre-authorization" of funds by the acquiring system – just like Uber "withdraws" your money before the trip starts, but it can credit it back instantly in case of rejection – however in this case, funds are exchanged between individuals. Tokens will only be frozen on the consumer's account until the users agree to complete the service transaction, whether it is successful or not.

3. Ethereum Blockchain

Finally, the use of the Ethereum Blockchain as an economic basis allowing you to make a reliable purchase. Below are several ways to use CARTOKEN. In future, its use could be extended to other transport-related services.

5.3 Infrastructure

Implementation
Parkapp Service Fee

Today there are plenty of mobile apps to pay for parking on the parking market of megacities, but there is still no unified system or unified platform to serve this huge market. Mobile apps are holding steady and classic tactics are being deployed in the battle for market share. They are heavily advertised and make money from parking payments by charging a user fee of up to 11% of the payment amount. Market players cannot charge a lower fee, since there are fixed costs to cover a centralized server, salaries, payment processing bank and the need to make at least some profit.

One of the aims of the CARTOKEN is to reduce or eliminate this fee and even provide with good discounts in the ParkApp ecosystem, both when paying for single sessions at commercial parking lots and when topping-up user monthly parking cards for public parking lots in all of the world's big cities.

The mobile app will have an internal wallet where the user will be able to store CAR tokens. When using for parking, toll roads, fast food drive-thru chains etc. with the mobile app, the user will have the option of reducing the service fee by using tokens.

What else

The ParkApp platform builds not only the mobile app, but also its own ecosystem that will include an interaction with the user's car. Automatic payment for on-street parking from the car's on-board computer, which was discussed earlier, also assumes a built-in fee for individuals and a subscription payment for corporate clients. These amounts can also be "paid" in just the same way using a CAR token. The user must have enough tokens on the balance and set up automatic one-click "payment" with tokens.

Important

Commercial properties consider to include the ParkApp fee into their rates, which for the user means there is no difference in the cost of service paid through a parking meter/machine or via ParkApp. Thus, the ability to use tokens will ultimately make service cheaper, and token owners who have no need for parking services will be able to sell tokens to drivers within ParkApp.

Implementation
Reserve a Spot: P2P Usage

The social module is designed for communications between the users and is not limited to the functionality of the ParkApp built-in messenger.
Currently, an internal Escrow agent is in development for networking with a decentralized P2P system which will give users (natural persons) an opportunity to earn tokens or make real money on:

Your own parking spaces (AirBnB model but for the parking space).
This may be either a garage in a private house or a place owned in a covered parking lot.

The purpose of this service is leasing a parking space when it is not used by its owner. An owner makes the parking space publicly visible in the mobile application. Now this parking space is displayed as a separate icon with a higher priority than the nearby parking spaces. A person who wants to park a car there transfers CAR for this space to its owner upfront.

In this P2P system, the ParkApp ensures payments, transfers, and refunds of tokens between users. All transactions and parking sessions are recorded in blocks and saved in a decentralized Blockchain via Smart contract. The Blockchain and Smart contract technologies ensure authenticity and safety of information on the transactions. No third party can falsify or modify this information.

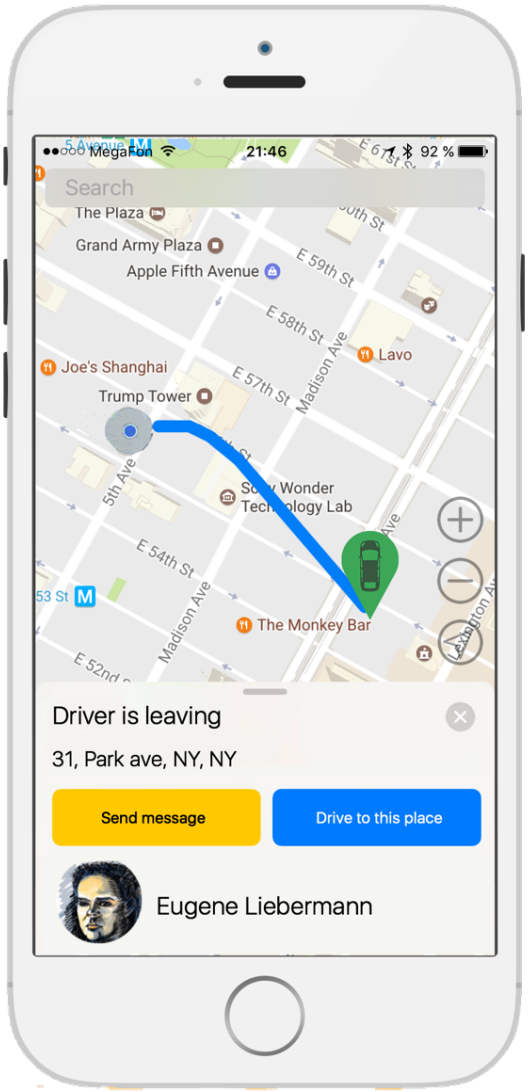
Your private time
The driver who needs a parking space transfers CAR to another, who occupies it for the time of waiting.

Example 1: you have an appointment at a location where a parking space is hard to find. The price of one hour parking doesn't matter: for you it is much more important to be on time and to leave your car without the risk of it being towed away. The driver who is about to leave the parking space in this area can wait for you, and you will transfer CAR to him for the time of waiting as agreed.

Each user has a crypto-wallet with a unique ID which shall be provided to the other user for transferring tokens to the account. In the ParkApp built-in messenger you can share this ID in an intuitive manner with the driver whom you are waiting for in order for them to transfer the agreed number of tokens to your account.

How does it work?

Driver A "Bob" is going to leave the parking spot in 15–20 minutes. In the mobile application with just a tap he makes the spot publicly visible to apply for.
Driver B "Dylan" can see it on the map as a blinking icon. By tapping on this icon, the user is able to text another one:



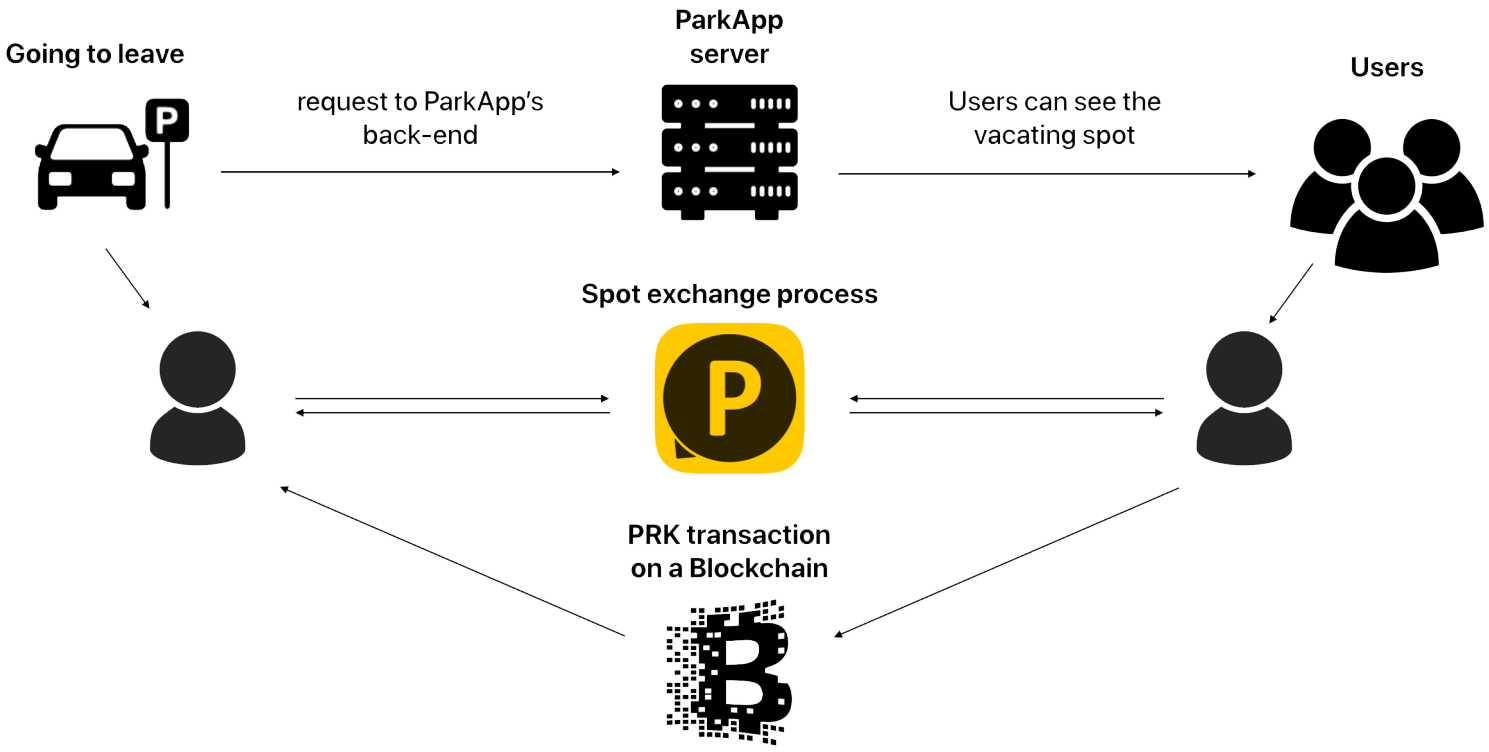
- **Dylan:** Hi Bob! I'd like to take your spot as I have no time on cruising around the location for another... Could you please hold it for me for 10 minutes for 30 CAR's!

- **Bob:** Deal! Follow the mark on the map and search for a red Ford Mondeo.

- **Dylan:** I am there please share your wallet ID so I could send you 30 CAR's.

Dylan just easily made a transaction to Bob and earned his rating placed on a Blockchain.

The “Take my Place” service has a rating system for registered and active users and will use the “genuine rating” technology. Use of this technology for a user rating system will mean that only another user who has participated in a two-way exchange about this parking space can leave a review and/or an assessment about the user. The 'genuine rating' system is based on the use of Blockchain technology, which guarantees the authenticity of the rating system. Participants in the rating system will be not only individual users, but also any parking facilities: street, parking lots, shopping mall parking lots, and airports.



Implementation
Discounts on Parking Payments

Use CAR token for the part of the cost.

We consider the markets of Asia, USA and Europe to be comfortable from the point of view of implementing the crypto-payments for the parking service. Those commercial parking lots that will accept CAR token for payment will have the highest rating and attendance in the system.

Using both Fiat money and CARTOKEN

The service charge (transaction fee) of any kind of payments could be made in CAR. We think this will be the predominant method of parking payment as far as it enables users to receive further discounts.

Discounts

Use tokens if the owner of a commercial parking lot provides a discount for ParkApp users according to the customer loyalty program.

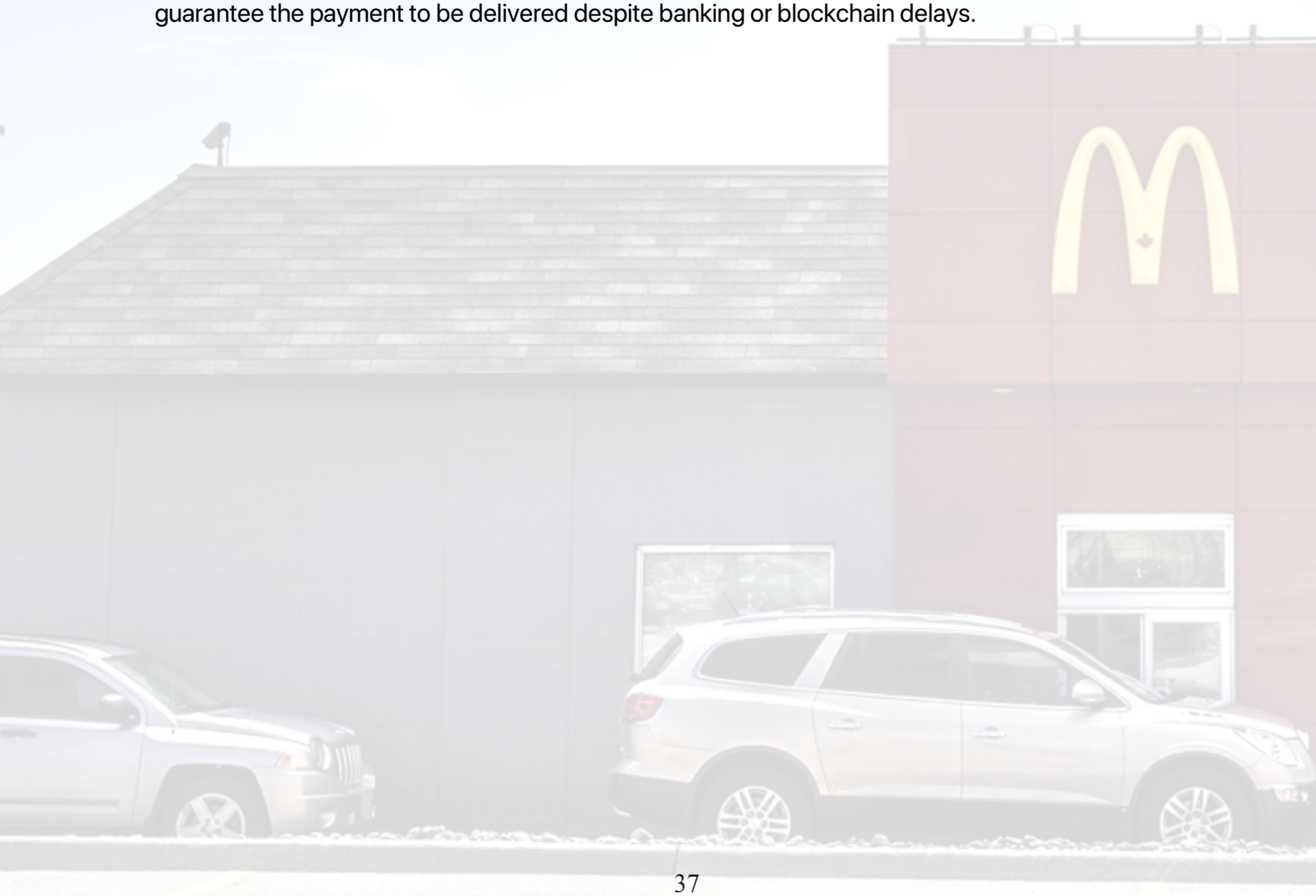
For example, to pay for 1 hour with fiat money, get 1 hour for tokens and get 1 hour as a bonus for free.

Implementation
Auto-Payments for drivers related service

Drive-Thru chains

McDonalds, Starbucks and other worldwide leaders of a fast food chains for drivers are the perfect example of the congestion and lines at the registers, effectively leading to a dissatisfaction and customer loss.

ParkApp Pay automatic payments from vehicles on-board are to make the service faster and guarantee the payment to be delivered despite banking or blockchain delays.



Crypto Payments

Peer-to-peer transactions on the roads and parking space

P2P car selling/buying for crypto

know previous owners using ParkApp Pay as an escrow of the deal

Purchasing new cars with crypto

avoiding difficulties and fees of fiatization

Car rentals

Car wash

Maintenance

Insurance

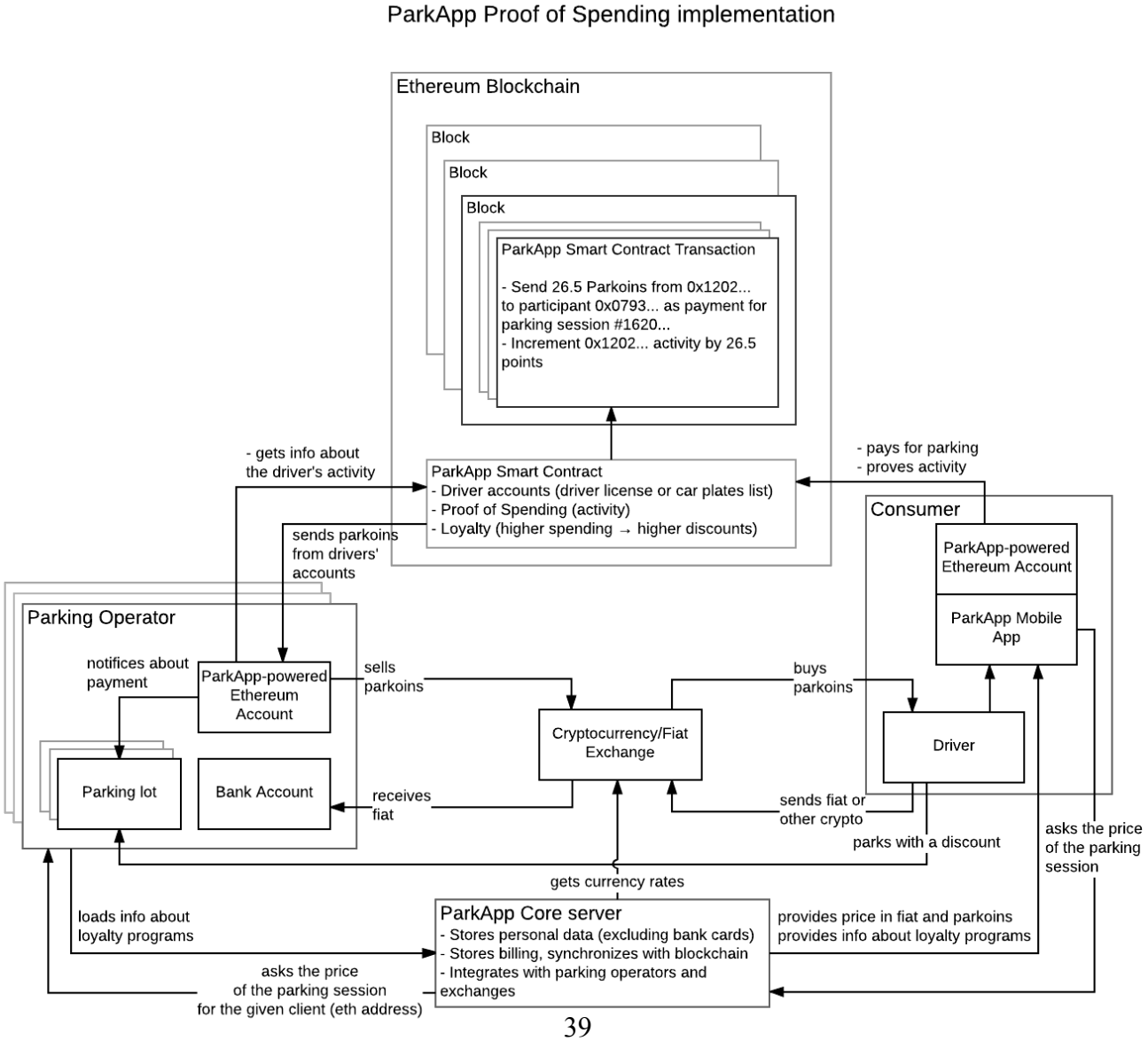
The Loyalty Program Infrastructure

1. The driver downloads the app, which works with the ParkApp server and Ethereum Blockchain. So, a person can store his private key both on his device and on the server, authorized via phone number.

2. The ParkApp user finds a parking lot and can find out the price per hour using the app with his personal loyalty program.

3. Before leaving (or while parking if the parking lot requires) the driver pays for his parking time using blockchain, which guarantees transparency of the history and genuineness of the driver's activity (identified by license plate number or the app key, which can be transferred to the incoming driver's device by NFC) for all parking lot operators.

4. Depending on the activity over a period of time, the operators can offer significant discounts to the ParkApp user because clients will be attracted who don't save on parking, so boosting the parking lot's profit even after the discounts.



6 TECHNOLOGY NOTES

6.1 Blockchain

The Blockchain technology is being commonly introduced at various social levels starting from the government to the level of ordinary people who just like buying coffee for ETH or BTC. High demand of the Blockchain is stipulated by the level of its security and authenticity of the data stored.

This technology is mainly based on decentralizing the stored information between all network participants and reliable crypto-protection algorithms for transactions to be conducted.

ParkApp is going to use Blockchain not only for the CARTOKEN, but also in the other areas that can help users to defend their interests. This includes challenging parking fines, which can sometimes be charged due to software or parking equipment malfunction. With a Blockchain, all user parking payment transactions will be securely stored, and ParkApp can stand surety for any parking payment or help challenge a fine.

1. Storage of user transaction data and further verification in case of disputes.
Some Blockchain implementations are given below:

Data to be stored:

- Parking start time
- Parking duration
- Exact geographical coordinates
- Zone number (if any)
- Payment method
- Amount paid
- Payment confirmation (bank transaction number)

2. Storage of service subscription data: auto-payments, subscriptions, confirmation of the user agreement for any service, etc.

3. If a user visits partner networks, he/she is awarded with CARTOKENs when paying for service. This information is recorded to the Blockchain as confirmed user activity history.

6.2 Data security

A lot of attention is placed on the security when running the Token Sale. The most reliable tools are used to provide cryptocurrency security.
The three pillars of security are: multisig, the Ledger cryptocurrency hardware wallet, and the team.

Multisig Cryptocurrency Wallet

Performing any operations with the collected funds requires multiple (2 of 3) confirmations from authorized members of our team, each of whom has a personal private access key. This eliminates the possibility of losing funds as a result of a loss of a single private key.

Ledger Cryptocurrency Hardware Wallet

- Is one of the most reliable cold-storage cryptocurrency hardware wallets. This means that private keys from authorized team members are stored on a physical device without a permanent connection to the Internet, which minimizes the possibility of a private key being stolen in a hacking attack. Moreover, this protection is multiplied by three.

The Team

The key to ParkApp security. A highly qualified staff of developers with previous experience in defending against various cyberattacks, and a team of advisers with support from advisors, confidently guarantee the security of our buyers' transfers. The trust and support expressed by the engineers and consultants from Ford Motor Company also point to the loyal intentions and the forward-thinking strategy of our team.

7 ROADMAP

7.1. High level road map

March-June 2018 — Token Sale preparation and conduction
- Legal preparation, Whitepaper
- Marketing campaign

June 2018 — Token Sale results processing
- Runway specification based on the of the Token Sale results
- Fiatisation and Development of the planned functionality

July 2018 — 'ParkApp Pay' Beta launch
- Public launch of an automatic parking payment platform for carmakers and municipalities

Sept 2018 — CARTOKEN wallet
- Integration of an ERC20 wallet into ParkApp mobile app
- Starting to accept the service fee in CARTOKEN

Oct 2018 — US Expansion
- Integration with US commercial parking corporations
- Launching USA marketing campaign

Nov 2018 — ParkApp Pay development
- Commercial ticket-less parking payments
- On-street parking payments

Dec 2018 — Peer to peer services
- Launch of peer to peer service

Jan 2019 — Further geographical expansion
- At this point we expect to have at least 2M US drivers as our active users
- Launching a marketing campaign in Canada, UK, Europe, Japan, China, South Korea

Mar 2019 — Integration with car-related services
- Integration with car washers, car services, toll roads, etc.

Jan 2020 — Integration with partners worldwide
- Integration with European and Asian municipal on-street parking and car services partners

Jan 2021 — Unified International tool and platform
- At this point we expect to have 25M users with daily CARTOKEN turnover of 120M CAR.

7.2 To be developed

Follow-up customization of the product and platform. Integration with a built-in crypto-wallet

A separate crypto-wallet is created for every platform: Server Side, Android OS, iOS. These implementations use native Ethereum API- and web3-based libraries for every platform.

The Android OS implementation uses the web3j library. With this library, developers can establish communication with Ethereum JSON-RPC over HTTP and IPC, which enables the support of an Ethereum wallet in light mode and a full range of features to work with smart contracts.

Using the built-in crypto-wallet

A private wallet designed exclusively for CARTOKEN (CAR) tokens is created automatically for every ParkApp user. This measure is intended to enhance the security of the entire ParkApp payment system, users' personal data and funds.

To work with their ParkApp wallet, users must transfer CAR tokens from their personal wallet (the one with CAR tokens purchased from the ParkApp Token Sale.

ParkApp does not store the private user keys on its server. All CARTOKEN transactions are performed via the mobile application. When a user checks a transaction status, the mobile application only sends the transaction number to the server; the server then retrieves the transaction details from https://etherscan.io.

HOW TO: Pay for the Services with Tokens

Payment for the services through ParkApp can be made in several ways depending on the type of a service (for more information about the services, see Section 3):

1. **Full amount with Fiat Money**

For example, payment for the private parking including commission can be made in the same currency as required by the parking operator. For instance, in the US dollars, if parking is located in San Francisco. The user might be interested in this type of payment, if there are no tokens on his/her account, or he/she is in hurry and not willing to wait for a refill.

2. Using both Fiat money and CARTOKEN

For example, payment for a private parking including commission can be made in CARTOKEN. We think this will be the predominant method of parking payment as far as it enables users to receive further discounts.

3. Full amount with CARTOKEN

For example, payment for peer-to-peer services such as "Hold a spot" and "Peer-to-peer taxi". It is also possible to pay with tokens if the owner of a commercial parking lot provides a discount according to the customer loyalty program. For example, to pay for 1 hour with fiat money, 1 hour with tokens and get 1 hour as a bonus for free. Then for the user and the owner it looks like a discount, and the CAR token does not lose its properties of the "Product / Utility token".

Ethereum limitations

Ethereum transactions are deemed fully credible after several confirmations from the other network participants to avoid "double expenses", i.e. the duplication of withdrawals. For the user, this means that after transfer of funds to another address he/she will have to wait 14 seconds to 2 minutes before they are reliably credited there.

As long as such delays are undesirable in terms of the ticketless parking payment system, in the case of a hybrid payment, we allow the user to receive confirmation without waiting for transaction confirmations from Ethereum if there has already been received a payment confirmation from the Bank for the main part of payment. However, in order to avoid fraud, we will block the possibility to pay for the services with tokens for those users whose transactions are not confirmed by the network within 5 minutes following the stated moment.

As for peer-to-peer services, delays in payment are not so critical here, thus, crypto transactions between the users will rely entirely on Ethereum Blockchain, communications with which will take place on the side of the mobile app.

ParkApp Pay

ParkApp contactless payment proprietary standard utilising embedded crypto wallet in tandem with NFC or Merchant server API via ParkApp servers.

Development of automatic parking payment is the priority.

ParkApp Pay workflow philosophy:

1. User signs up for ParkApp, adds the car make, model and OLP (official license plate), and ties his/her bank card through bank acquiring.

2. A user arrives to a parking lot that supports automatic payment for ParkApp Pay. The parking server recognises the license plate number and requests the ParkApp server to verify activation of the auto-payment service according to specific license plate number.

3. The ParkApp server verifies activation of the ParkApp Pay and validity of the tied credit card; if successful, it starts a parking session for that user.

4. When the user is ready to leave he/she can merely approach the exit from the parking lot, where the license plate is re-recognised and a payment query is made. The user leaves the lot after a successful payment without doing anything but driving own car.

Big data

We intend to use neural network based algorithms to analyse a bunch of groups.
1. Traffic data, including those received from the application users. This will enable the drivers to always have at hand the real-time information on parking lot availability.
2. Data from CCTV cameras to determine availability of on-street parking lots and location of free spaces therein.

Decentralized Statistical Data Analysis System

The development of a decentralized and effective system for analysing photo data directly on the users' mobile devices. In contrast to a centralized system for processing data in the cloud, a decentralized system uses distributed mobile device resources, the performance of which in the last few years has reached the required level for neural network processing.

This approach makes maximum use not only of the device's central processing unit, but also of its graphical processing unit, which has even more processing power. A parking lot photo undergoes the entire processing procedure on a mobile device: from initial processing to neural network processing and parking parameter recognition.

Geographical expansion to USA market

Delivering entirely functional platform
Registration of a legal entity in the USA;
Integration with our partners companies on the east and west coasts;
Product Launch;
Marketing campaign.

The promotion is focused on the "Parking Module" (vacant spots and leaving drivers in a real time), payments with the parking session extension and automatic parking guide with navigation.

Launching the ParkApp service on the top 100 cities in US in terms of population, also integrating with SAP connected parking to integrate the platform with more than 2.5 million parking spaces.

To reach the 10 m users audience ParkApp should involve the following advertisement tools:

Facebook, Instagram, Twitter, Pinterest targeted daily ads (mobile advertising platforms).
Google AdWords, YouTube and Vimeo videos showing the benefits of ParkApp functionality.
AppStore and Play market search optimization (showing ParkApp first on "parking" related requests.
App landing page and app indexing.
AppStore and Google play market featuring. Co-branding with car rentals.
Cooperation and co-branding with carmakers.
Front desks of hotels, instructions on the mall parking area.
On-line publications (IT, cars, transportation, mobile apps review sites).
External ads, banners on streets and subway.

The retention ratio: 60% of margin is back to the marketing and ads.

Preliminary charging policy (includes the bank fee):

If the transaction is $35 or less - we charge 9% in CAR,
If the transaction is $36-50 - we charge 8% in CAR,
If the transaction is $51 or more - we charge 7% in CAR.

8. TOKEN SALE MANUAL

Instructions

Detailed instructions on each step can be found in the "Instructions for Participation" section at the end of the Whitepaper.

Register an ERC20 compatible wallet like 'MyEtherWallet'.
If you have a wallet, sign in to your account.

Top up your wallet with Ether (ETH) to buy CAR.

Send the amount of ETH to the Smart-Contract address placed under the 'GET TOKEN' button on the landing page https://parkapp.io.

Make sure that the Ethereum network has confirmed your transfer at the web-site https://etherscan.io.

To participate you need an ERC20-compatible wallet, such as **Mist**, **MetaMask**, **MyEtherWallet** or **Parity**.

Please note that currency exchanges, as well as Exodus and Jaxx wallets are not compatible. Participants who try to use these methods will not be able to receive tokens.

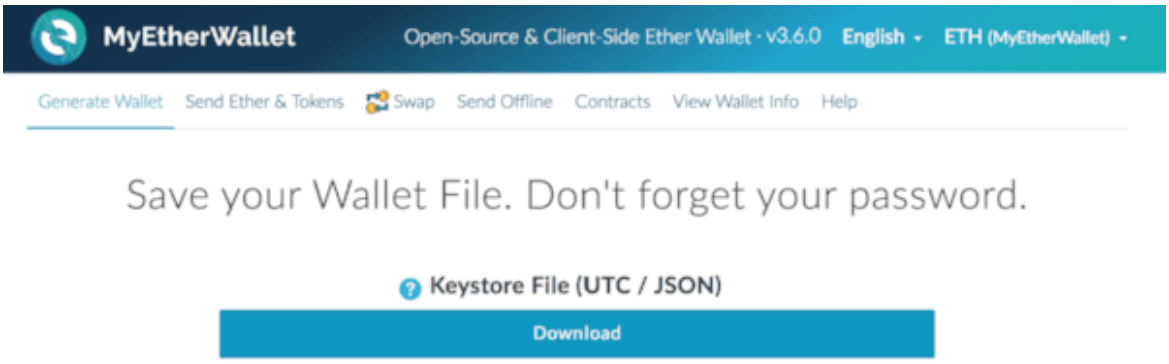
Basic instructions for sending ETH for each wallet:

- MyEtherWallet – Official
- MetaMask – Official
- Parity - Informal, recommended by the Parity team
- Mist - Unofficial

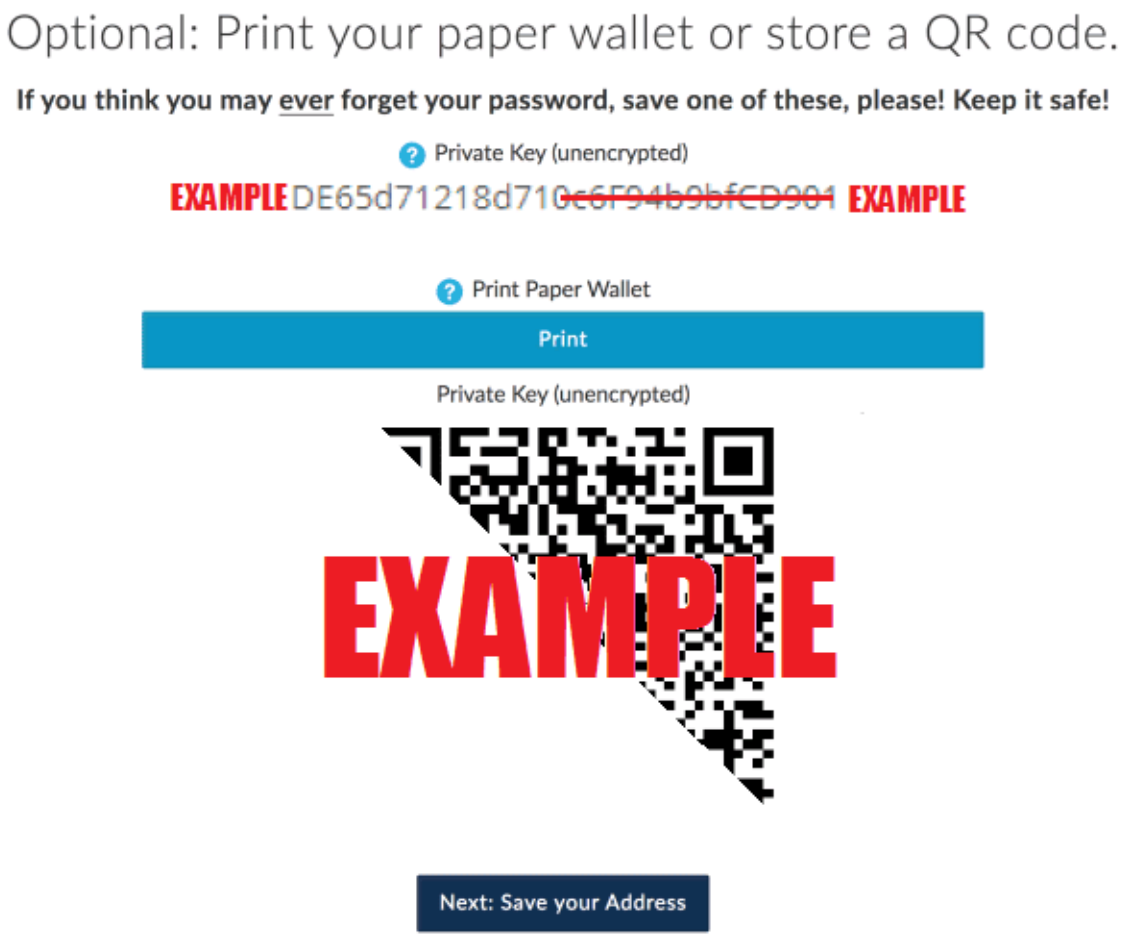
Creating an account in 'MyEtherWallet'

1. https://myetherwallet.com
2. Create a wallet with a password and save the password in your password manager

3. Download the key file. Keep it in a safe place and do not give it to anyone. You will need it to use your MyEtherWallet account. Do not lose and do not transfer this file to anyone and do not show the password from it, because they are controlled access to your tokens. This file you can use with other Ethereum wallets.

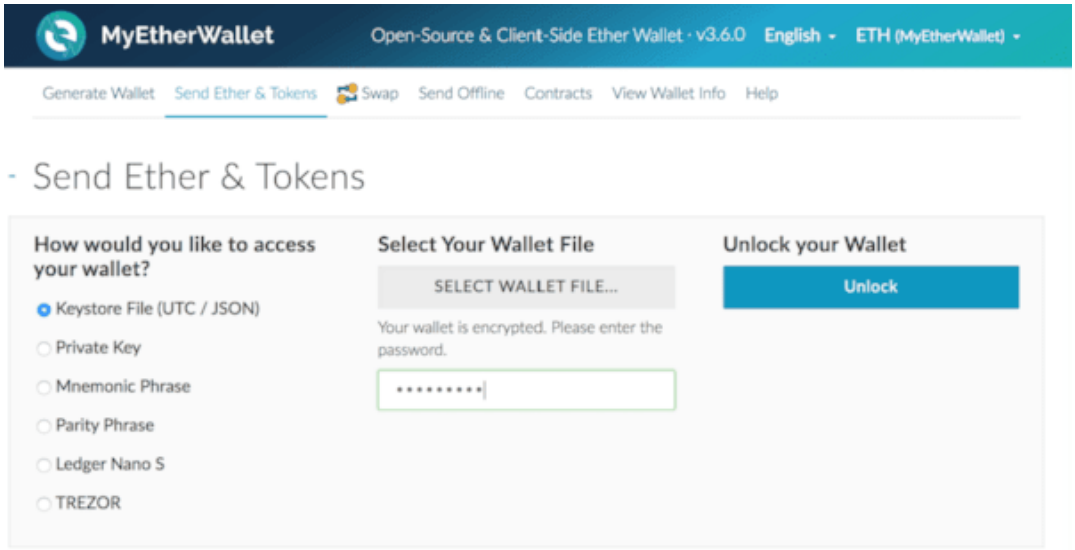


After downloading the key, you will be offered to print a paper backup, which we also recommend you to do.

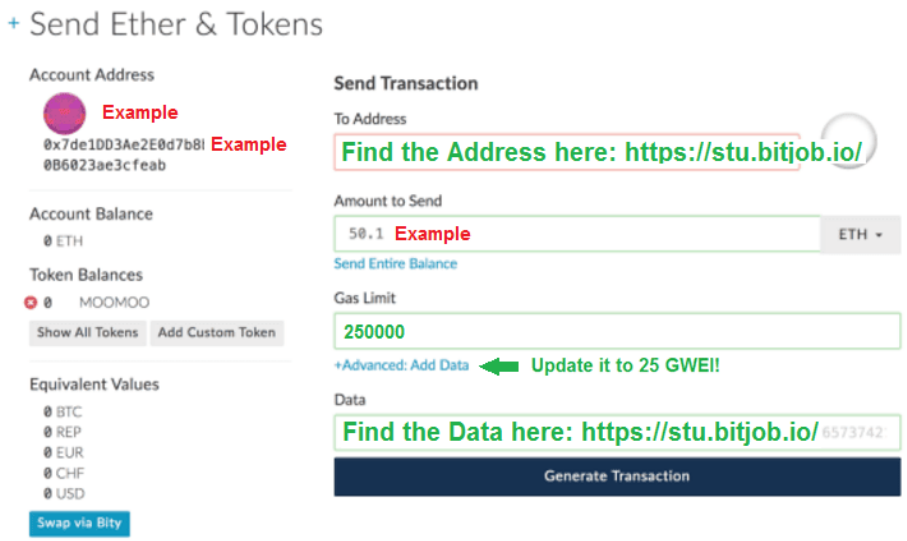


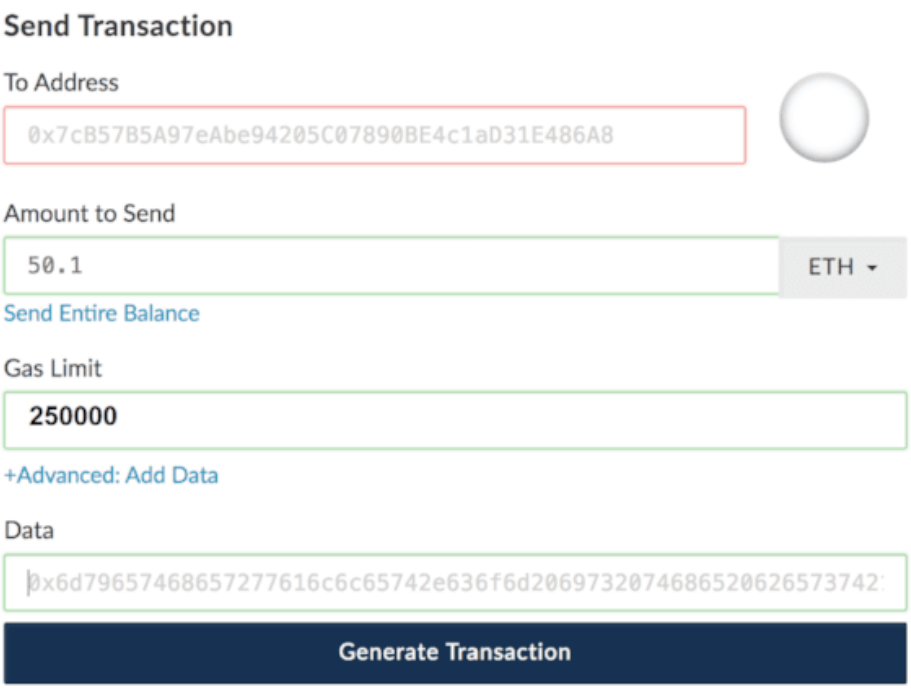
Top up the wallet

Go to MyEtherWallet using the Send Ether & Tokens link. Send your key file and password.



Send ETH from your exchange account to MyEtherWallet. Transactions on the Ethereum network are usually confirmed up to 2 minutes. If the ETH balance is not visible, refresh the page in the browser and login into the wallet again.





Transfer Ether

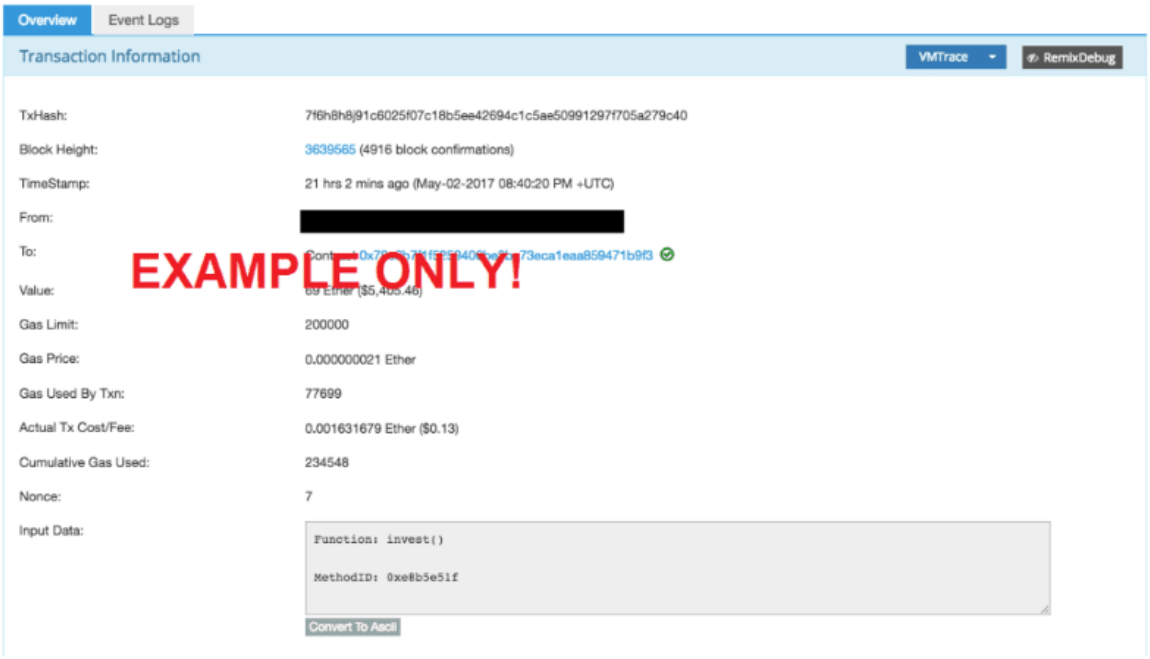
When you have a balance on your personal wallet, you can make a transfer to the sales contract with the specified parameters. You must specify all the options, otherwise the smart contract will not accept your transfer.

In case of an incorrect Gas limit or input data, your ETH will be immediately sent back.

Checking the transaction success

After the transaction, MyEtherWallet will give you a link to Etherscan.io to track your transaction in the green dialog box. Click on the link and make sure that you have a green tick. If so, then your transfer to the sales contract was successful.

Below is an example of a successful transaction:



TEAM

Eugene Liebermann
CEO, founder ParkApp

4 years of robotics and AI development for civil use only.
5 years in Smart-City and parking space industry development.
Smart-city, smart-parking expert.

Nadya Che
CMO
Business development director
15 years in banking system and sales

Al Diaz
ambassador in NY, USA
Business development director in US
8 years in sales

Sean Johnson
Southern US
municipal relations
Mississippi, USA

Alexander Huseu
CTO
7 years in Android OS development.
5 years in secure VoIP calls and messaging.
Ford Applink (SYNC) developer from ParkApp end.
Node.js, JavaScript, Python, Solidity and TensorFlow.

Baev Sergey
System architect
Full stack Rails developer, DevOps
8 years in WebDev, 6 years in Ruby on Rails,
Lead webdev on 10+ production-released projects

Max Ryzhkov
iOS developer
7 years in iOS Mobile development

Eugene Pavlyuk
7 years in iOS Mobile development,
3 years in Secure Communication,
Data Protection and Encryption.
Ford Applink (SYNC) developer from ParkApp end
Munich, Germany

PARTNERS & ADVISORS

Oleg Gusihin
Ford Motor Company HQ
Detroit, MI, USA

Technical Leader,
Inventor of the Ford on-board
Interactive system with external apps

Max Gritsenko
Ford Motor Company

Senior IT project manager
Technical, Marketing expert

Natalie Karayaneva
CEO at Propy
Menlo Park, CA, USA

Software Engineer
Oxford Graduate
Smart City expert

Alexander Menn
Mentor, Smart-City,
Smat-Parking expert
in Europe and Russia.

Vlad Isenbaev
GM Cruise LLC
Hacking and cyber security expert
San Francisco, CA, USA

David Drake
Advisor, US investors and PR community
Founder and Chairman of LDJ Capital, a family office
New York, NY, USA

Jim Blasko
Blockchain Advisor,
Founder of UNB coin
Las Vegas, NV, USA

CONTACT US

https://parkapp.io

- https://blog.parkapp.io
- https://twitter.com/park_app
- https://www.facebook.com/parkapp.io
- https://t.me/ParkApp

iOS app
https://appsto.re/ru/jEll5.i

Android app
Https://play.google.com/store/apps/details?id=com.parkingsolutions