**# P2P-Lending**

**## Table of Contents**

- [About the Project](#about-the-project)

- [Challenges](#challenges)

- [Built With](#built-with)

- [Getting Started](#

- [Prerequisites](#prerequisites)

- [Installation](#installation)

<!-- ABOUT THE PROJECT -->

**## About The Project**

This repository hosts a variety of smart contracts for a DAO (Decentralized Autonomous Organisation) focussing on delivering a decentralized p2p-lending platform on the Ethereum Blockchain. This may very well develop into something completely different based on the direction taken by the DAO in the future. We are as excited as you are! ;-)

\_A decentralized, smart contract based platform for p2p-lending on the Ethereum Blockchain can play the role of a bank in the process of lending money to one or more beneficiaries.

The open ecosystem of the p2p-lending platform has the potential to offer cheaper lending contracts than traditional centralized institutions, while also enabling people all over the world to profit from a fair and transparent portfolio of products.\_

**### Challenges**

To solve multiple challenges within such a substantial system, we are conductiong extensive research into Ethereum and other potential smart contract platforms as well as continue developing a modular system based on components solving those challenges one by one. The most pressing challenges to solve for such a system include:

- Proof of Identity

- Decentralized management and progression of the ecosystem by a DAO

- Staying within the bounds of Ethereum

- Incentivization of the DAO ecosystem and its continuous use

- Building financially profitable and inclusive products

- Accessibility and Usability of the p2p-lending plattform and ecosystem

**### Built With**

- [Node 10](https://nodejs.org/en/)

- [Truffle](https://truffleframework.com/truffle)

- [Vue.js](https://vuejs.org/)

- [web3.js](https://web3js.readthedocs.io/en/1.0/getting-started.html)

<!-- GETTING STARTED -->

**## Getting Started**

To get a local copy up and running follow these steps.

**### Prerequisites**

These are the requisites you need, in order to use the software and instructions, on how to install them.

- [Node.js](https://nodejs.org/en/):

**- macOS:**

- Install [Homebrew](https://brew.sh/):

/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

- **Update Homebrew:**

brew update

**- Check if Homebrew is ready to brew:**

brew doctor

- **Install Node.js:**

brew install node

- **Test that Node.js was installed by running:**

node –version

**- Windows:**

- **Install [Chocolatey](https://chocolatey.org/install):**

**- via cmd.exe:**

@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command "iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))" && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"

**- via PowerShell.exe:**

Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))

**- Install Node.js:**

choco install nodejs

- **Test that Node.js was installed by running:**

node --version

- [npm](<https://www.npmjs.com/>)

**- Install npm:**

npm install npm -g

**- Test that npm was installed by running:**

npm --version

- [Ganache](https://truffleframework.com/ganache)

- [Truffle](<https://truffleframework.com/truffle>)

- **Install Truffle:**

npm install truffle -g

**- Test that truffle was installed by running:**

truffle version

**### Installation**

**1. Clone the repo**

We might have to upload this on our github to be able to clone out.

git clone

**2. Run Ganache on port 8545**

Ganache -> Settings -> Server -> Port Number -> 8545

**3. Install dependencies**

npm install

**4. Compile Smart Contracts**

truffle compile

**5. Deploy Smart Contracts to local blockchain**

npm run migrate:dev

**6. Switch to frontend folder**

cd frontend

**7. Install frontend dependencies**

npm install

**8. Start frontend**

npm start

**9. Open the DApp in your favorite browser**

localhost:8080