

APRIL 28, 2021



Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

Decentralised Application for Crowdfunding on a Blockchain

USER MANUAL

Student: Karl Whelan
Supervisor: Geoff Hamilton
Student Number: 15561423

Contents

1. Introduction	2
2. Installation.....	3
2.1. Hardware requirements	3
2.2. Software requirements	3
2.2.1. Compiling and deploying smart contracts locally	3
2.2.2. Compiling and deploying smart contracts to a public blockchain.....	4
2.2.3. Installing and building the user interface	4
2.3. Compiling and deploying the smart contracts.....	4
2.3.1. Deploying to a local Ganache blockchain	5
2.3.2. Deploying to public Rinkeby blockchain	6
2.3.3. Deploying to other public blockchain	9
2.4. Installing and building the user interface	9
2.4.1. Installing Metamask	9
3. Testing.....	10
3.1. Testing the user interface.....	10
3.2. Unit testing and end-to-end testing for smart contracts	11
3.3. Scalability and gas usage testing for smart contracts	12
4. Usage	14
4.1. App layout	14
4.1.1. Landing page	14
4.1.2. Home page.....	17
4.1.3. Create project page	18
4.1.4. View projects page	18
4.1.5. Help Page	20
4.2. Using the App.....	21
4.2.1. Getting test ether	22
4.2.2. Creating a project	24
4.2.3. Donating to a project.....	25

1. Introduction

This Document outlines the steps involved in installing, configuring, using, and testing this application. To compile, deploy, test, and configure this project you need to be in a Linux environment. This application can be deployed to the Ethereum Mainnet, any of its test networks or a locally running Ganache blockchain.

Once deployed live on a blockchain the user interface can be used in any browser that supports the Metamask extension. The currently supported browsers are Chrome, Firefox, Microsoft Edge and Brave. The UI can also be used on mobile however mobile browsers do not currently support extensions so a mobile user must download the Metamask App and use that apps browser to open the UI.

A live version of the user interface connected to the smart contracts deployed on the Rinkeby testnet can be found here:

<https://main.d3oz5l3o8ahlw9.amplifyapp.com/>

To use this application, you must have Metamask installed. For more information see the UI landing page or the installation section below.

2. Installation

The following installation guide is for Linux operating systems. The project can be cloned with HTTPS using:

```
:~$ git clone https://gitlab.computing.dcu.ie/whelak26/2021-ca400-whelak26.git
```

2.1. Hardware requirements

To connect to the smart contracts that are already deployed on the Rinkeby testnet or to compile and deploy to a local Ganache blockchain there are no special hardware requirements beyond a Linux operating system.

To compile and deploy the contracts yourself to the Ethereum Mainnet or one of its test networks you will need to be running a local node of that blockchain on your machine. To run a local node of a blockchain requires a large amount of free Solid State Drive space. The amount of space needed will vary by network and over time. Bellow is an estimation of the requirements for a local full node for the Rinkeby testnet:

- 50G+ SSD Storage
- 4G+ Ram also recommended.

2.2. Software requirements

Below is a list of the software requirements for the various stages of installation along with directions to download the software. I have broken them down into subheadings, so the user need only download the software for what they are trying to achieve.

2.2.1. Compiling and deploying smart contracts locally

- Truffle:

```
:~$ npm install -g truffle
```
- Ganache. Install the self-contained prebuilt binary located here:
<https://www.trufflesuite.com/ganache>

2.2.2. Compiling and deploying smart contracts to a public blockchain

- Truffle:

```
:~$ npm install -g truffle
```
- Go Ethereum (Geth). See the installation guide below for how to install Geth:
<https://geth.ethereum.org/docs/install-and-build/installing-geth>

2.2.3. Installing and building the user interface

- Node and npm:

```
:~$ sudo apt install node.js
```
- Metamask. Add the Metamask extension to the browser of your choice:
<https://metamask.io/download.html>

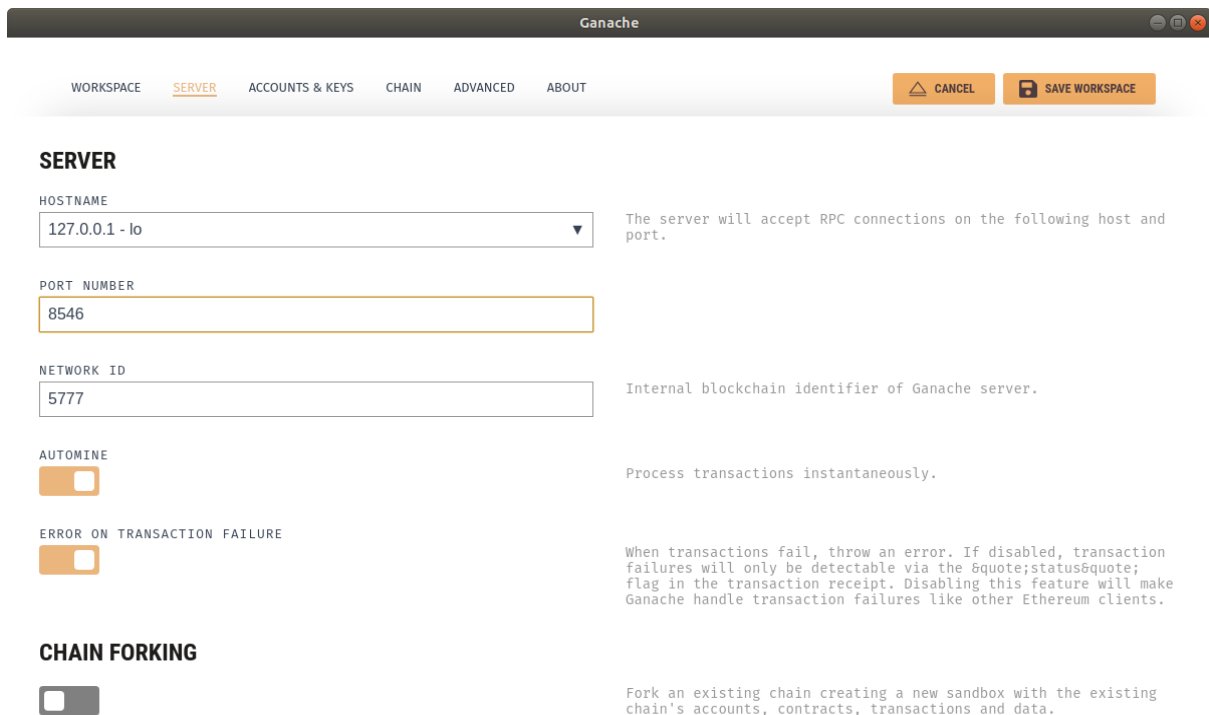
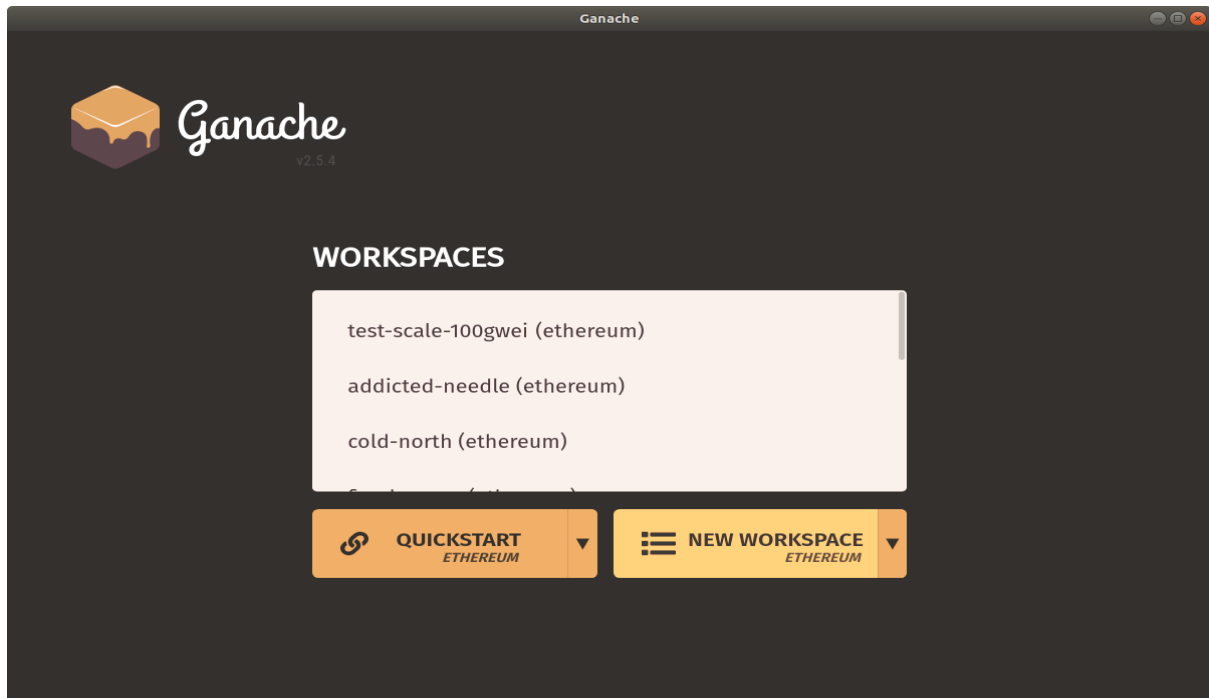
2.3. Compiling and deploying the smart contracts

To compile the smart contracts using truffle navigate to the /src folder and use:

```
:~/ca400/2021-ca400-wheelak26/src$ truffle compile
```

2.3.1. Deploying to a local Ganache blockchain

Open the Ganache application and run a local blockchain on port 8546. What you choose for the other settings does not matter.



SERVER

HOSTNAME
127.0.0.1 - lo

PORT NUMBER
8546

NETWORK ID
5777

AUTOMINE
☒

ERROR ON TRANSACTION FAILURE
☒

CHAIN FORKING
☐

The server will accept RPC connections on the following host and port.

Internal blockchain identifier of Ganache server.

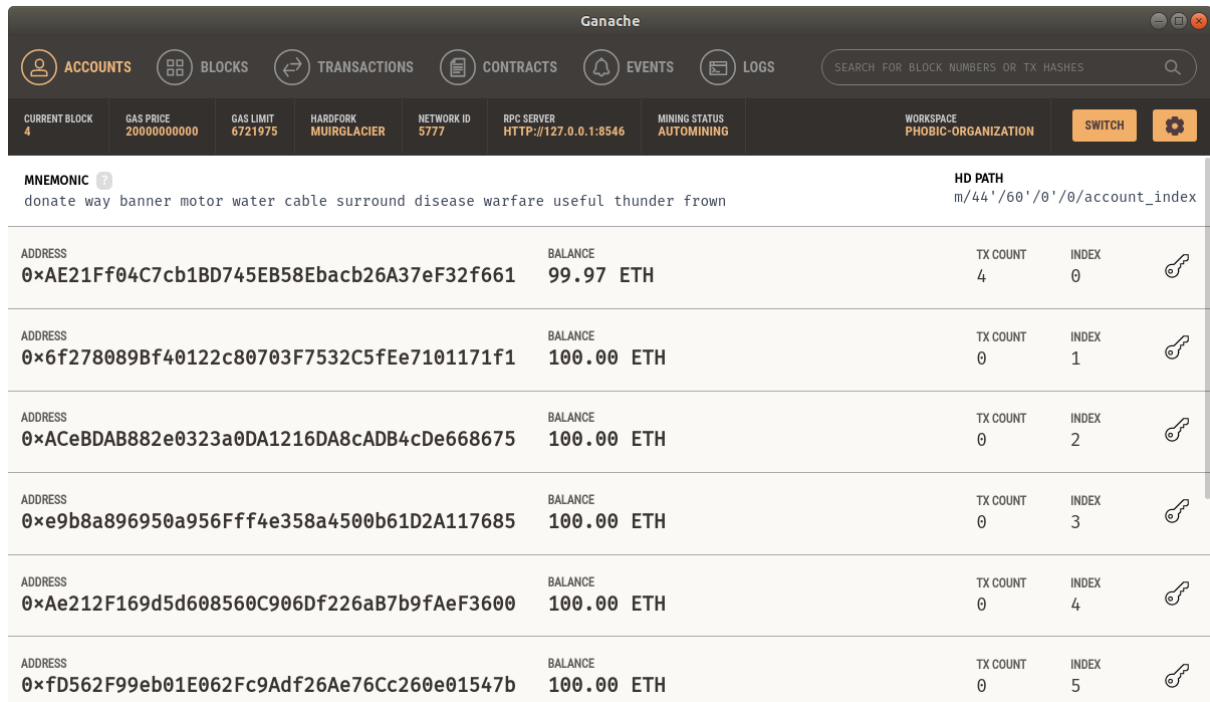
Process transactions instantaneously.

When transactions fail, throw an error. If disabled, transaction failures will only be detectable via the `status` flag in the transaction receipt. Disabling this feature will make Ganache handle transaction failures like other Ethereum clients.

Fork an existing chain creating a new sandbox with the existing chain's accounts, contracts, transactions and data.

This will also generate a list of accounts, by default 10 accounts, which contain some test ether. Using the following command will deploy the smart contracts to the local blockchain using the first account in the list:

```
~/ca400/2021-ca400-whelak26/src$ truffle migrate --network development
```



Ganache				
ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS				
SEARCH FOR BLOCK NUMBERS OR TX HASHES				
CURRENT BLOCK 4	GAS PRICE 20000000000	GAS LIMIT 6721975	HARDFORK MUIRGLACIER	NETWORK ID 5777
RPC SERVER HTTP://127.0.0.1:8546			MINING STATUS AUTOMINING	
WORKSPACE PHOBIC-ORGANIZATION			SWITCH ⚙️	
MNEMONIC donate way banner motor water cable surround disease warfare useful thunder frown		HD PATH m/44'/60'/0'/0'/account_index		
ADDRESS 0xAE21Ff04C7cb1BD745EB58Ebacb26A37eF32f661	BALANCE 99.97 ETH	TX COUNT 4	INDEX 0	🔑
ADDRESS 0x6f278089Bf40122c80703F7532C5fEe7101171f1	BALANCE 100.00 ETH	TX COUNT 0	INDEX 1	🔑
ADDRESS 0xAceBDAB882e0323a0DA1216DA8cADB4cDe668675	BALANCE 100.00 ETH	TX COUNT 0	INDEX 2	🔑
ADDRESS 0xe9b8a896950a956Fff4e358a4500b61D2A117685	BALANCE 100.00 ETH	TX COUNT 0	INDEX 3	🔑
ADDRESS 0xAe212F169d5d608560C906Df226aB7b9fAeF3600	BALANCE 100.00 ETH	TX COUNT 0	INDEX 4	🔑
ADDRESS 0xfD562F99eb01E062Fc9Adf26Ae76Cc260e01547b	BALANCE 100.00 ETH	TX COUNT 0	INDEX 5	🔑

2.3.2. Deploying to public Rinkeby blockchain

To deploy to the Rinkeby testnet you must have a local node running on your machine at port 8545. To see how to connect yourself to Rinkeby see here:

<https://www.rinkeby.io/#geth>

You'll also need to create an ethereum account and password:

```
$ geth account new
```

```
Your new account is locked with a password. Please give a password. Do not forget this password.
Password:
Repeat password:

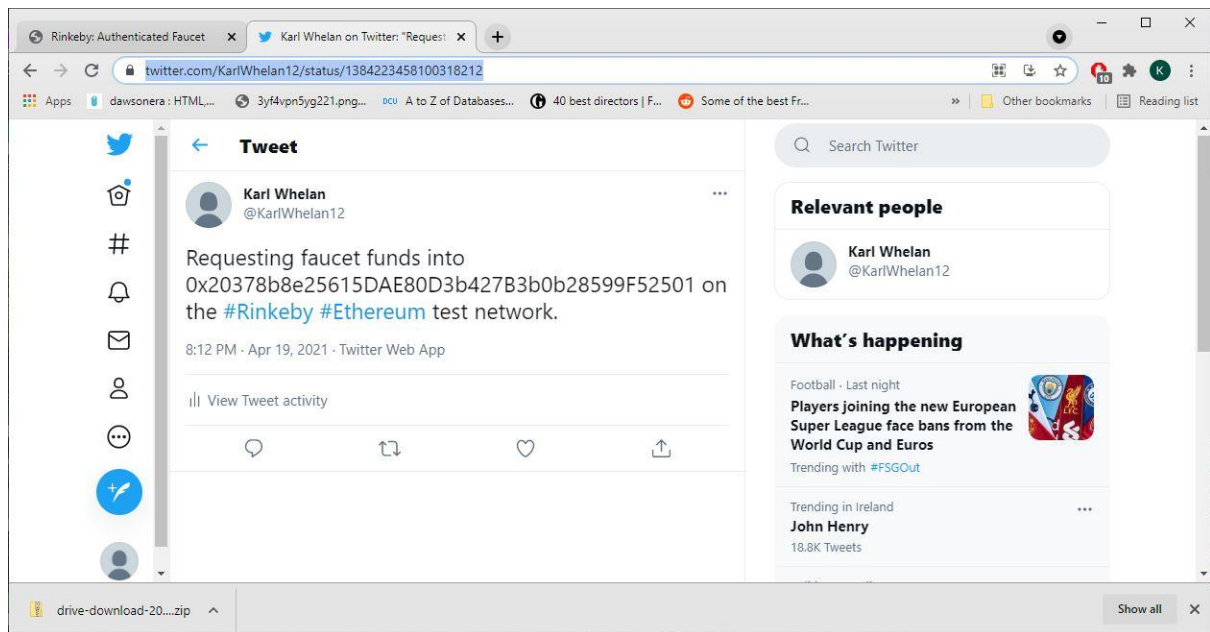
Your new key was generated

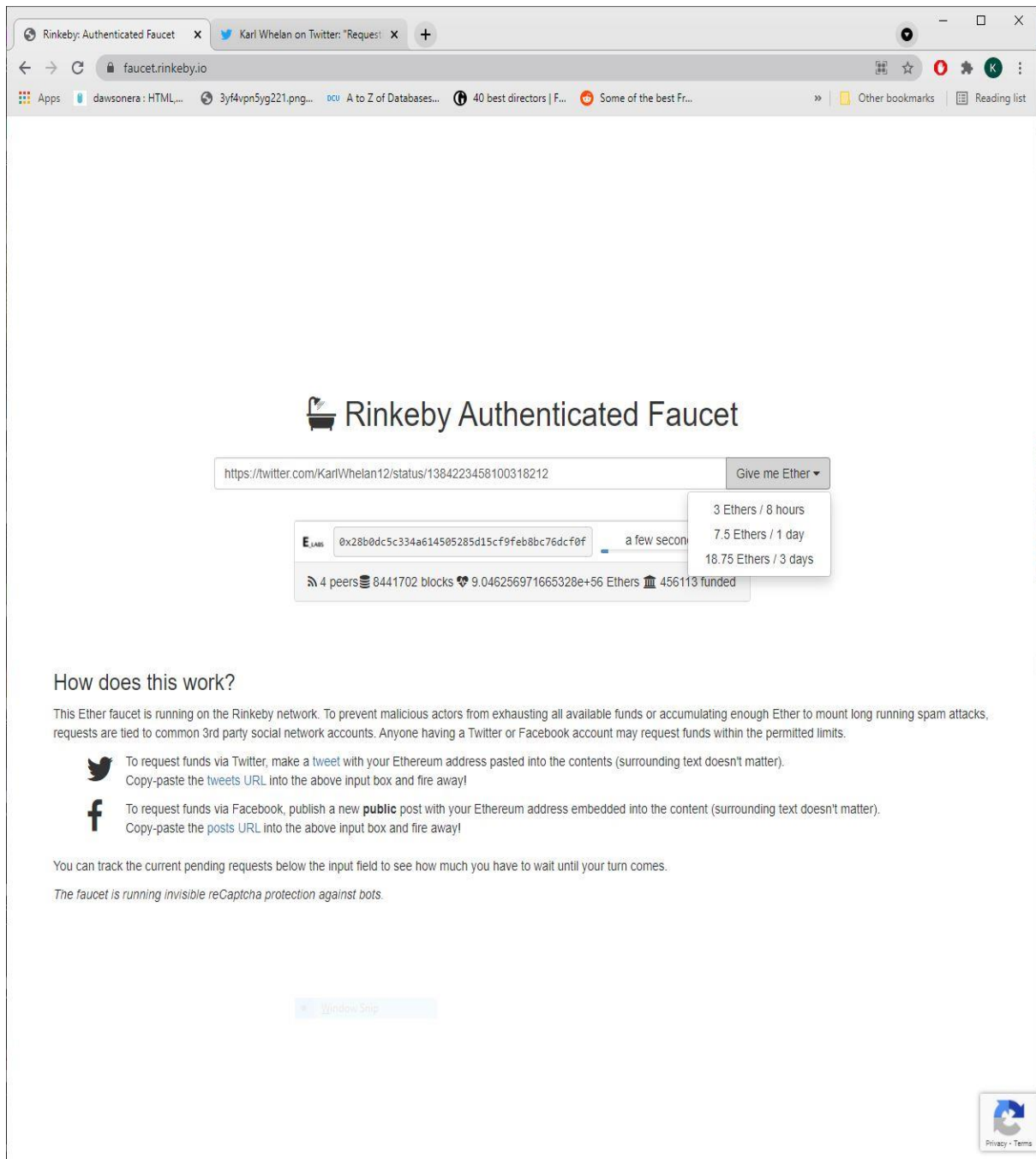
Public address of the key: 0x20378b0e25615DAE80D3b427B3b0b28599F52501
Path of the secret key file: /home/karl/.ethereum/keystore/UTC--2021-04-19T18-52-18.712901157Z--20378b0e25615dae80d3b427b3b0b28599f52501

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!
```

Now that you have your new account. You can use the public key to request funds from the Rinkeby faucet. You will need these funds to deploy the contracts. To do this you need to make a public media post containing your accounts public key, save the URL of that post and paste the URL into the Rinkeby faucet here as shown in the images below:

<https://faucet.rinkeby.io/>





Once you have test ether in your account you should unlock the account with your password and use truffle to deploy to the Rinkeby testnet as below:

```
~/ca400/2021-ca400-whelak26/src$ truffle migrate --network Rinkeby
```

2.3.3. Deploying to other public blockchain

This project can also be deployed, with truffle, to the Ethereum Mainnet or any of its test networks by configuring the truffle-config.js file in /src. For more info about how to do this and other configurations like changing the ports used see the truffle docs:

<https://www.trufflesuite.com/docs/truffle/reference/configuration>

2.4. Installing and building the user interface

Compiling the smart contracts will create a folder /src/client/src/builtContracts containing json files that the user interface needs to connect to the blockchain. To install the user interface, navigate to the /src/client folder and use:

```
~/ca400/2021-ca400-whelak26/src/client$ npm install
```

To run the app in development mode use:

```
~/ca400/2021-ca400-whelak26/src/client$ npm start
```

This will run the app here: <http://localhost:3000/>

To build the app for production use:

```
~/ca400/2021-ca400-whelak26/src/client$ npm run build
```

This will build the app, so it is ready for production and place the build files in /src/client/build.

2.4.1. Installing Metamask

To use the UI, you must have the Metamask extension installed on your browser:

<https://metamask.io/download.html>

3. Testing

3.1. Testing the user interface

Each component in the frontend has tests associated with it that test that it renders correctly given certain conditions. There are also end-to-end tests that test the navigation elements of the UI. For the end-to-end tests you must first have a local ganache blockchain running on port 8546. To run all tests for the UI navigate to `/src/client` and use:

```
~/ca400/2021-ca400-whelak26/src/client$ npm test
```

This will open the console. Type 'a' to run all tests. You should receive feedback of all tests passing as shown below.

```
No tests found related to files changed since last commit.  
Press 'a' to run all tests, or run Jest with '--watchAll'.
```

Watch Usage

- › Press a to run all tests.
- › Press f to run only failed tests.
- › Press q to quit watch mode.
- › Press p to filter by a filename regex pattern.
- › Press t to filter by a test name regex pattern.
- › Press Enter to trigger a test run.

```
PASS  src/components/__test__/Navbar.test.js  
PASS  src/components/__test__/ViewProject.test.js  
PASS  src/components/__test__/MetamaskInfo.test.js  
PASS  src/components/__test__/ViewProjectsPageBody.test.js  
PASS  src/components/__test__/CreateProjectPageBody.test.js  
PASS  src/components/__test__/GettingFundsInfo.test.js  
PASS  src/components/__test__/HomePage.test.js  
PASS  src/components/__test__/Input.test.js  
PASS  src/components/__test__/Header.test.js  
PASS  src/components/__test__/AboutPage.test.js  
PASS  src/components/__test__/ToolTipIcon.test.js  
PASS  src/App.test.js
```

```
Test Suites: 12 passed, 12 total  
Tests:       34 passed, 34 total  
Snapshots:   0 total  
Time:        2.038 s  
Ran all test suites.
```

3.2. Unit testing and end-to-end testing for smart contracts

To run the Unit tests and end-to-end tests for the smart contracts you must also have a local Ganache blockchain running on port 8546. Navigate to /src and use:

```
:~/ca400/2021-ca400-whelak26/src$ truffle test ./test/projects.js
```

You should receive feedback of what tests are running and passing like below.

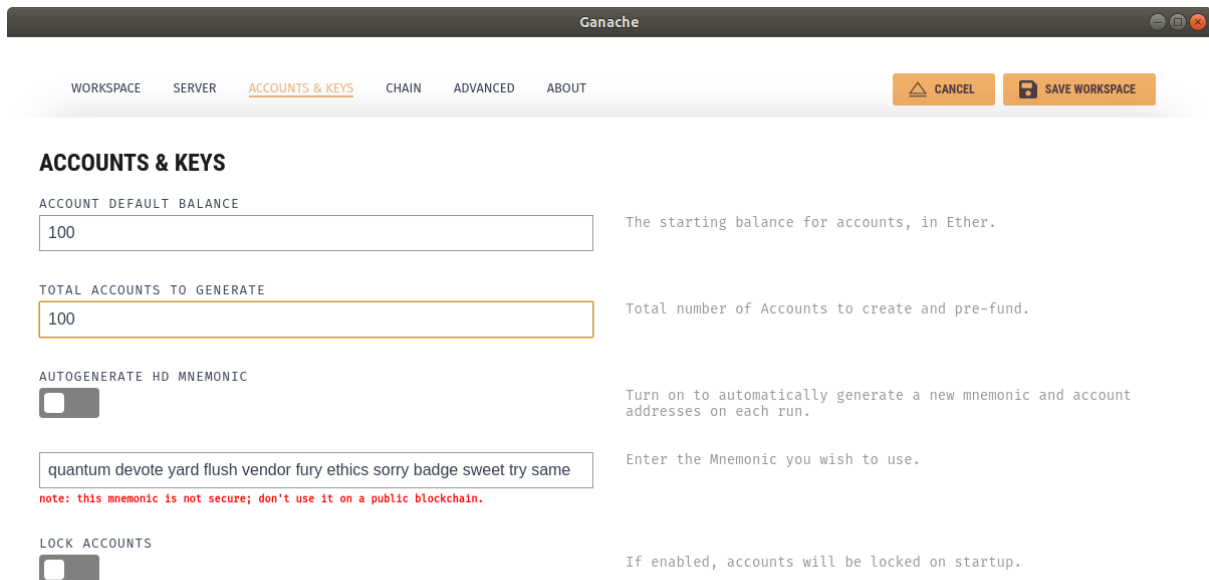
```
Compiling your contracts...
=====
> Everything is up to date, there is nothing to compile.

Contract: Projects
  ✓ should deploy the contract properly
  ✓ Should create a Project correctly (73ms)
  ✓ Should create 3 more Projects correctly (204ms)
  ✓ Should return 4 projects in the correct format (77ms)
  ✓ Should donate 1 ether to each project (349ms)
  ✓ Should check that the balance of the contract is 4 eth
  ✓ should advance time
  ✓ Should create a project that ends in two days (109ms)
  ✓ Should advance time by 2 days and the project should have ended
  ✓ Should test payOut function (139ms)
  ✓ Should test payRefunds function (361ms)

11 passing (2s)
```

3.3. Scalability and gas usage testing for smart contracts

For the scalability and gas usage testing you should run a local Ganache blockchain with 100 accounts, instead of the default 10, on port 8546.



ACCOUNTS & KEYS

ACCOUNT DEFAULT BALANCE
100
The starting balance for accounts, in Ether.

TOTAL ACCOUNTS TO GENERATE
100
Total number of Accounts to create and pre-fund.

AUTOGENERATE HD MNEMONIC
☐
Turn on to automatically generate a new mnemonic and account addresses on each run.

quantum devote yard flush vendor fury ethics sorry badge sweet try same
Enter the Mnemonic you wish to use.
note: this mnemonic is not secure; don't use it on a public blockchain.

LOCK ACCOUNTS
☐
If enabled, accounts will be locked on startup.

Navigate to /src and use:

```
~/ca400/2021-ca400-whelak26/src$ truffle test ./test/scalability.js
```

This should give you feedback in the console like below. It also will write the results of the testing to 3 separate files in the /src/test/logs folder. The tests measure gas usage when calling functions multiple times. If the gas usage can be shown to be linear then the functions are scalable.

```

Compiling your contracts...
=====
> Everything is up to date, there is nothing to compile.

```

Contract: Scalability Scalability test

- ✓ tests creating 100 projects (5884ms)
- ✓ tests donating 99 times to 1 project (10701ms)
- ✓ tests donating 1 time to 99 projects (16519ms)

3 passing (33s)

create-project-function-scalability.log						
	transactionNumber	GasUsed	GasPrice	Exchange rate	Cost in Ether	Cost in Euro
1	0	132641	1	1870	0.000132641	0.24803867
2	9	117641	1	1870	0.000117641	0.21998867
3	19	117641	1	1870	0.000117641	0.21998867
4	29	117641	1	1870	0.000117641	0.21998867
5	39	117641	1	1870	0.000117641	0.21998867
6	49	117641	1	1870	0.000117641	0.21998867
7	59	117641	1	1870	0.000117641	0.21998867
8	69	117641	1	1870	0.000117641	0.21998867
9	79	117641	1	1870	0.000117641	0.21998867
10	89	117641	1	1870	0.000117641	0.21998867
11	99	117641	1	1870	0.000117641	0.21998867
12	99	117641	1	1870	0.000117641	0.21998867

donate-to-project-function-scalability.log						
	transactionNumber	GasUsed	GasPrice	Exchange rate	Cost in Ether	Cost in Euro
1	1	109776	1	1870	0.000109776	0.20528112
2	9	79776	1	1870	0.000079776	0.14918112
3	19	79776	1	1870	0.000079776	0.14918112
4	29	79776	1	1870	0.000079776	0.14918112
5	39	79776	1	1870	0.000079776	0.14918112
6	49	79776	1	1870	0.000079776	0.14918112
7	59	79776	1	1870	0.000079776	0.14918112
8	69	79776	1	1870	0.000079776	0.14918112
9	79	79776	1	1870	0.000079776	0.14918112
10	89	79776	1	1870	0.000079776	0.14918112
11	99	79776	1	1870	0.000079776	0.14918112
12	99	79776	1	1870	0.000079776	0.14918112

donate-to-all-project-function-scalability.log						
	transactionNumber	GasUsed	GasPrice	Exchange rate	Cost in Ether	Cost in Euro
1	1	64776	1	1870	0.000064776	0.12113112000000001
2	9	109788	1	1870	0.000109788	0.20530356
3	19	109788	1	1870	0.000109788	0.20530356
4	29	109788	1	1870	0.000109788	0.20530356
5	39	109788	1	1870	0.000109788	0.20530356
6	49	109788	1	1870	0.000109788	0.20530356
7	59	109788	1	1870	0.000109788	0.20530356
8	69	109788	1	1870	0.000109788	0.20530356
9	79	109788	1	1870	0.000109788	0.20530356
10	89	109788	1	1870	0.000109788	0.20530356
11	99	109788	1	1870	0.000109788	0.20530356
12	99	109788	1	1870	0.000109788	0.20530356

4. Usage

This section describes the layout and functionality of the User Interface. A live version connected to the Rinkeby testnet blockchain can be found here:

<https://main.d3oz5l3o8ahlw9.amplifyapp.com/>

4.1. App layout

4.1.1. Landing page

When you open the Application, you will be brought to the landing page which displays information about the App and gives detailed instructions on how to install Metamask if you have not already.

About

This project is a decentralised application(Dapp), implemented using blockchain technology and smart contracts, which acts as a crowdfunding platform. This platform allows creators with new ideas for projects to advertise these projects to the communities that may then fund them.

Crowdfunding is currently a popular way for creators to raise money but there are trust issues with the way most platforms are run that a blockchain can help solve. Most crowdfunding projects do not deliver on time and some never deliver at all. Every crowdfunding platform also charges fees, between 3 and 5% of all raised funding, and they act as a middle man you must trust to handle the transfer of funds appropriately.

When crowdfunding on a blockchain every transaction is transparent and traceable, which helps with trust, and smart contracts with predefined rules manage the transfer of funds so there's no middleman to trust, and no charges for using the platform. The blockchain network is also large, decentralized and encrypted protecting it from malicious attackers and single points of failure. This level of security is important when managing people's money.

This application is currently a proof of concept in that it is a fully functional platform but it is running on a test network, (the Rinkeby Testnet), where all the funds pledged and donated are fake. This allows users to interact with the platform without spending real money. The projects section is currently populated with example projects (taken from kickstarter) to give an idea of what the application will look like running on the Ethereum mainnet and populated with actual projects.

Instructions

To use this application you must have the Metamask extension installed on your web browser.

If you have Metamask Installed

The extension should pop up allowing you to connect with an account. This may take a few seconds to load.

If you do not have Metamask installed please follow these instructions

Metamask is available for Chrome, Firefox and Microsoft Edge.

[Chrome Link](#)

[Firefox Link](#)

[Edge Link](#)

Step 1

Click the link relative to you and add the Metamask extension to your browser. When prompted select the option to create a new wallet.

If you do have Metamask installed it should pop up now allowing you to sign in, choose your account and to connect to the blockchain.

main.d3oz5I3o8ahlw9.amplifyapp.com

dawsonera : HTML... 3yf4vpn5yg221.png... A to Z of Databases... 40 b

About

This project is a decentralised application(Dapp), implemented using blockchain technology which acts as a crowdfunding platform. This platform allows creators to fund their projects to the communities that may be interested.

Crowdfunding is currently a popular way for creators to raise money. However, most platforms are run that a blockchain can help solve. Most crowdfunding platforms never deliver at all. Every crowdfunding platform also charges fees, but they act as a middle man you must trust to handle the transactions.

When crowdfunding on a blockchain every transaction is transparent and smart contracts with predefined rules manage the transfer of funds so the creator doesn't need to use the platform. The blockchain network is also large, decentralized and secure from attackers and single points of failure. This level of security is inbuilt into the network.


This application is currently a proof of concept in that it is a fully functional decentralized application on the Rinkeby Testnet, where all the funds pledged and donated are in the form of test tokens. The projects section is currently a placeholder (from kickstarter) to give an idea of what the application will look like when it is launched with actual projects.

Instructions


To use this application you must have the Metamask extension installed on your web browser.

If you have Metamask Installed

The extension should pop up allowing you to connect with an account. This may take a few seconds to load.



Rinkeby Test Network



Welcome Back!

The decentralized web awaits

Password

.....

UNLOCK

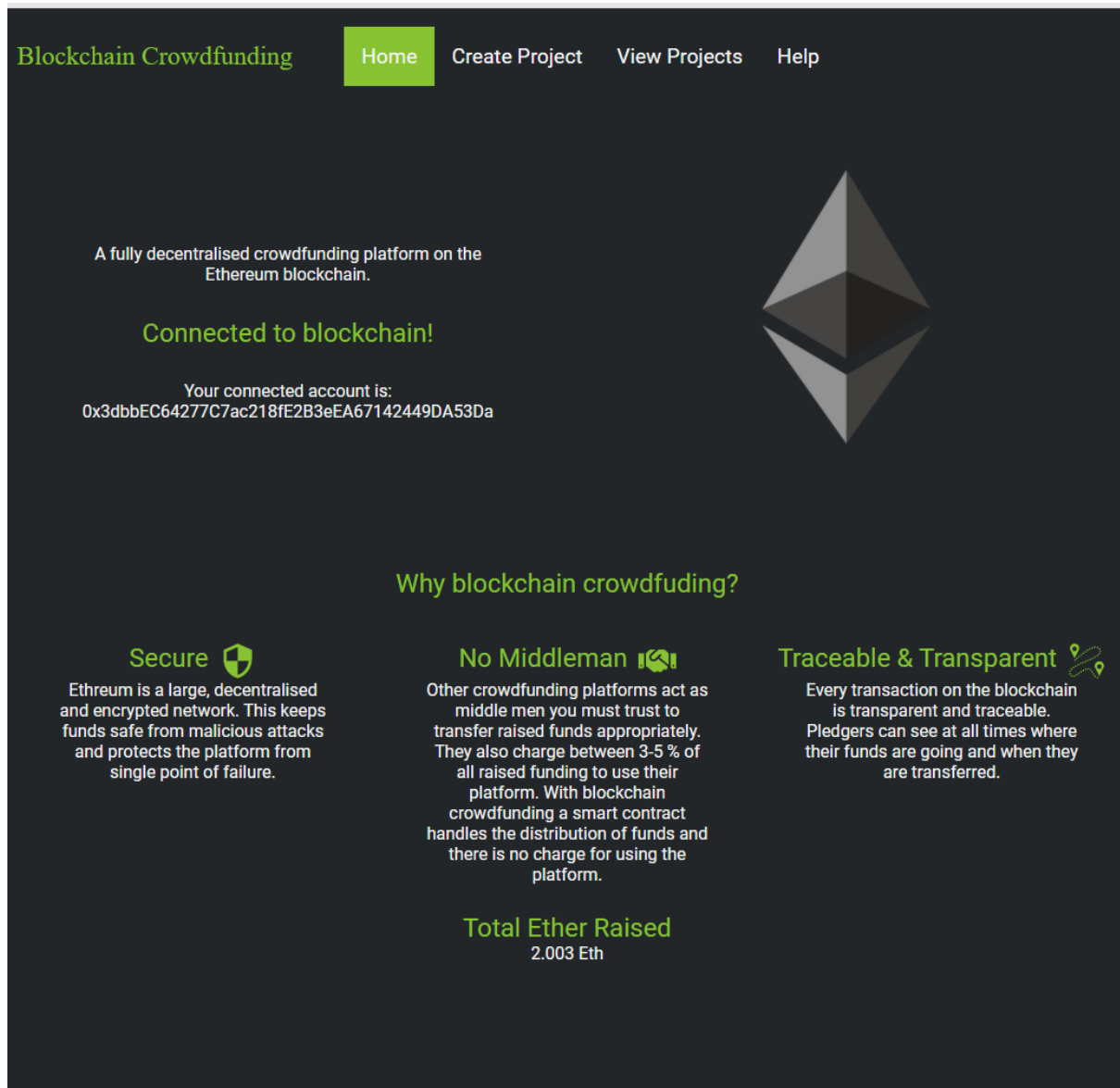
Restore account?

Import using account seed phrase

16 | Page

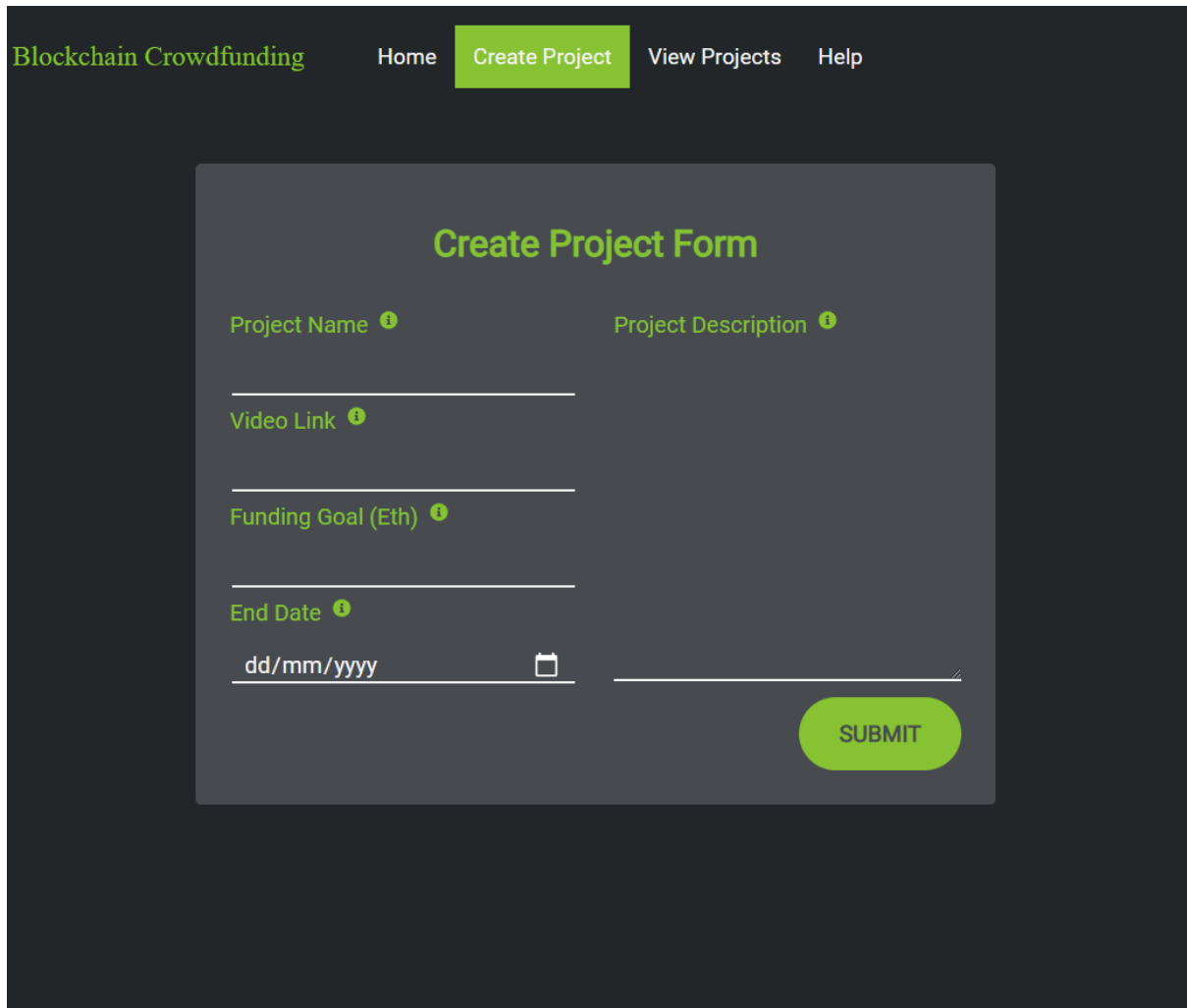
4.1.2. Home page

Once signed in you will be redirected to the homepage which contains some more information about the app including what account you are connected with and the total amount of Eth raised so far for active projects on the platform. You can now use the Navbar to navigate between pages.



4.1.3. Create project page

The create project page contains a submittable form that allows users to create a project. Tooltip icons give more detailed descriptions of the forms fields if the user needs them.



The screenshot shows a web application with a dark blue header. The header contains the text 'Blockchain Crowdfunding' in white, followed by navigation links: 'Home', 'Create Project' (highlighted with a blue background), 'View Projects', and 'Help'. Below the header is a light gray form titled 'Create Project Form' in bold black text. The form contains five input fields, each with a blue tooltip icon: 'Project Name', 'Project Description', 'Video Link', 'Funding Goal (Eth)', and 'End Date'. The 'End Date' field includes a date picker icon and a placeholder 'dd/mm/yyyy'. A blue rounded rectangular button labeled 'SUBMIT' is positioned at the bottom right of the form.

4.1.4. View projects page

The view projects page contains information about all active projects on the platform. The information on each project includes the project name, description, project video, funding goal, amount raised so far and the end date of the project. There is also a section at the bottom of each project allowing you to donate to that project should you wish. You can scroll through the projects by clicking the green arrows or by dragging across the screen with your mouse or finger if on mobile.

Mobi Hybrid Active Noise Cancelling 100 hr. Wireless Earbuds

Project Video



Project Description

Mobi is the first true wireless headphones to deliver full-frequency noise cancellation for a full-immersion audio experience. Mobi's hybrid ANC technology features 3 separate feedback and feedforward microphones on each earbud to capture and cancel noise at any frequency. Mobi actually hears what you hear and uses AI to create the ideal listening experience based on your music, your environment, and your movements.

The result? Full immersion into your music, videos, calls, and more. No distraction. No distortion. No disruption.

Combined with oversized speakers, an ultra-long battery, and IPX4 weatherproofing, Mobi can transform anywhere and everywhere into your own world of crystal-clear audio.

A revolutionary pair of earphones, MOBI is an enhancement to all your deep listening experiences. No more 'so-called' noise cancellation, MOBI is a guaranteed set of earbuds that will transform your view of noise cancellation with these unique features: AI powered Hybrid Active Noise Cancellation, 12mm Oversized Drivers for stronger bass, 100-Hour Battery Life and Multi Touch controls.

Mobi's AI Powered Hybrid Active Noise Cancellation is the newest and most advanced noise cancelling technology on the market.

Hybrid active noise cancellation combines feedforward and feedback microphones to capture a wider audio frequency, therefore cancelling more noise.

Funding Goal

5 Eth

Amount Pledged

1.003 Eth

End Date

6/6/2021

Donation Amount

4.1.5. Help Page

The help page contains information about the App, information on how to use the site and information about how to get test ether from the Rinkeby faucet.

Blockchain Crowdfunding Home Create Project View Projects **Help**

About

This project is a decentralised application(Dapp), implemented using blockchain technology and smart contracts, which acts as a crowdfunding platform. This platform allows creators with new ideas for projects to advertise these projects to the communities that may then fund them.

Crowdfunding is currently a popular way for creators to raise money but there are trust issues with the way most platforms are run that a blockchain can help solve. Most crowdfunding projects do not deliver on time and some never deliver at all. Every crowdfunding platform also charges fees, between 3 and 5% of all raised funding, and they act as a middle man you must trust to handle the transfer of funds appropriately.

When crowdfunding on a blockchain every transaction is transparent and traceable, which helps with trust, and smart contracts with predefined rules manage the transfer of funds so there's no middleman to trust, and no charges for using the platform. The blockchain network is also large, decentralized and encrypted protecting it from malicious attackers and single points of failure. This level of security is important when managing people's money.

This application is currently a proof of concept in that it is a fully functional platform but it is running on a test network, (the Rinkeby Testnet), where all the funds pledged and donated are fake. This allows users to interact with the platform without spending real money. The projects section is currently populated with example projects (taken from kickstarter) to give an idea of what the application will look like running on the Ethereum mainnet and populated with actual projects.

How to use this site

You can navigate to the various pages of this site to get a feel of the frontend for this application. If you have test funds in your Metamask wallet you can try creating a project or donating to an existing project to get a feel of how that works. For instructions on how to get test funds see below.

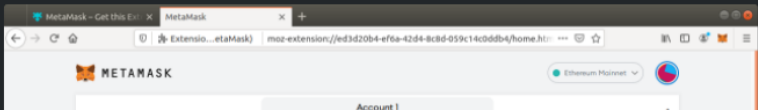
How to get test ether

If you want to test out creating a new project or donating to an existing one you will need to get some test ether. To do this you can either request ether from the rinkeby faucet or from the creator of this application. The faucet will deliver funds instantaneously but requires you to make a social media post requesting funds. Recieving ether from the applications creator will take longer but there is no need to make a social media post.

Using the faucet

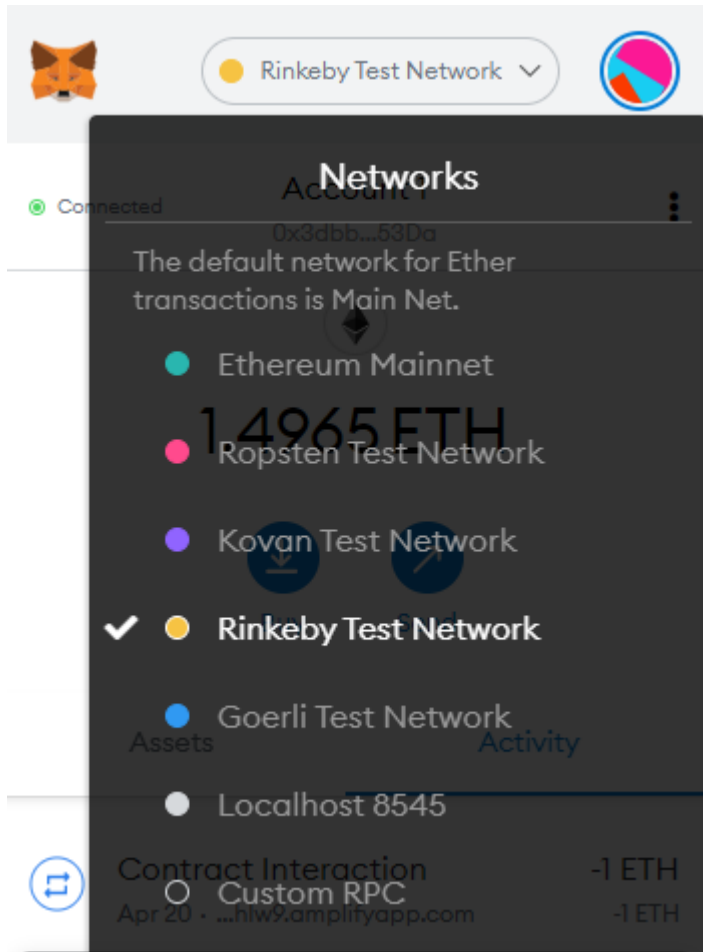
Step 1

Copy your account address from metamask as shown below.

A screenshot of a web browser window with the Metamask extension installed. The browser's address bar shows a URL starting with 'moz-extension://'. The Metamask extension interface is visible, displaying the Metamask logo and the text 'Account 1'.

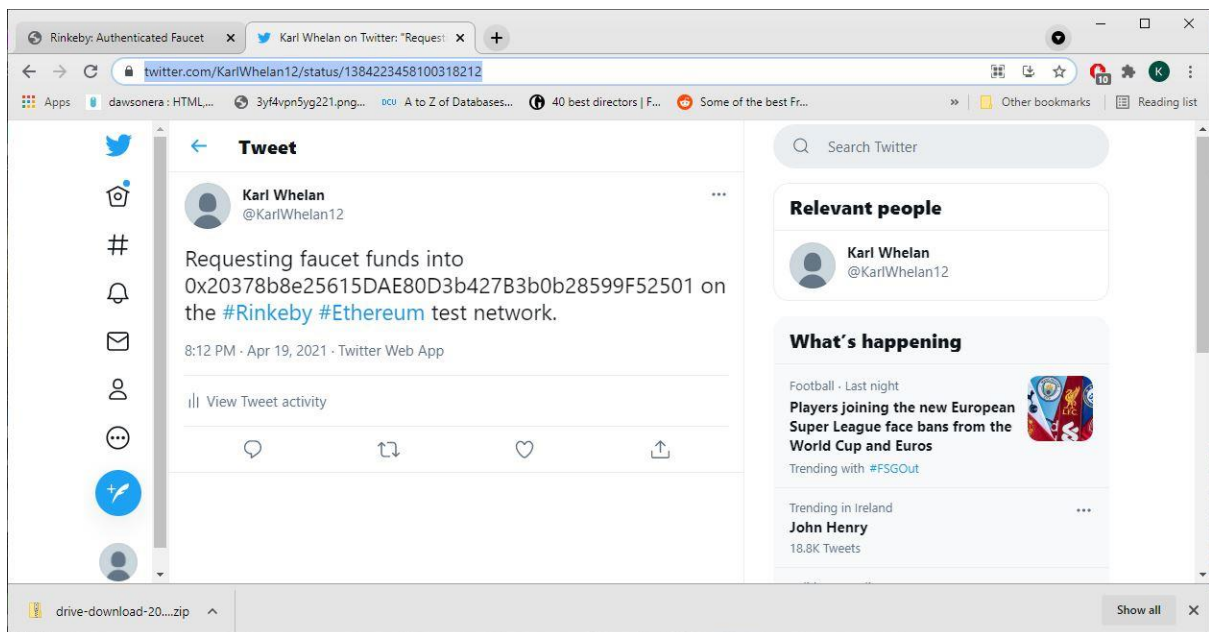
4.2. Using the App

Before using the App, please make sure you are connected to the correct blockchain on Metamask otherwise the functions will not work. If you are using the live UI from earlier in this section, you should be on the Rinkeby testnet as shown below.



4.2.1. Getting test ether

You will need Ether to create projects or to donate to existing projects. If you are using the live UI from earlier in this section, you will need Rinkeby test ether. To get some you can use the Rinkeby faucet. First make a public social media post, on twitter or Facebook, containing your public account address. Then go to the Rinkeby faucet [here](#) and paste the URL of you post into the faucet selecting the amount of ether you wish to receive. You should receive the funds in a matter of seconds. And example is shown below.





4.2.2. Creating a project

To create a project, fill in the create project form and press submit. You will not be able to submit the form if one or more of the fields are not valid. The tool tip icons can help you understand the various fields. When you have submitted valid values Metamask should pop up and allow you to accept or reject the transaction.

The screenshot shows the 'Create Project Form' on the Amplify app interface. The form includes the following fields:

- Project Name**: Sample Name
- Project Description**: This is a test descri
- Video Link**: <https://youtu.be/C8YbGa8epEQ>
- Funding Goal (Eth)**: 10
- End Date**: 29/05/2021

A green **SUBMIT** button is located at the bottom right of the form.

Overlaid on the right side of the form is a Metamask transaction confirmation window for the Rinkeby Test Network. It shows the following details:

- Account 1**: 0x18C8...6c...
- Contract Interaction**: <https://main.d3oz5l3o8ahlw9.amplifyapp.com>
- Gas Fee**: 0.000255 (No Conversion Rate Available)
- Gas Price (GWEI)**: 1
- Gas Limit**: 254580
- Amount + Gas Fee**: 0.000255
- Total**: 0.000255 (No Conversion Rate Available)

At the bottom of the Metamask window are **Reject** and **Confirm** buttons.

On the public testnet blocks take around 15 seconds to mine, after this you will receive a confirmation and your project will be added to the list of active projects.

4.2.3. Donating to a project

To donate to a project simply enter the amount of Eth you wish to donate in the donation field and click the button. Metamask will pop up with information about the proposed transaction and you can accept or reject it.

The screenshot shows a web browser window with the URL `main.d3oz5l3o8ahlw9.amplifyapp.com`. The page title is "A Kickstarter Project We Love: Soapbottle". The main content area features a video player showing various SOAPBOTTLE products (orange, pink, green, blue) and a "Watch on YouTube" button. Below the video is a "Project Description" section with the following text:

Meet SOAPBOTTLE, a new line of personal care that comes in packaging made from soap. Once the liquid content is finished, the empty container can be used as hand soap and "washes away" over time. Waste can be completely avoided.

This innovation rethinks everyday items and takes your experience with personal care to a new, enjoyable and meaningful level – without compromising sustainability.

We are launching our first product line here on Kickstarter, The SOAPBOTTLE Originals. They contain 100 ml of a soft and nourishing body wash made from natural ingredients.

In the future, we strive to create SOAPBOTTLES in various colors, shapes, and scents. Besides, it is our goal to offer a great variety of personal care products, such as shampoo, shower gel and more.

The problem: Every year, millions of tons of packaging waste are sent to landfills or disposed of into the environment, leading to the drastic degradation of the natural ecosystems on which we depend on. Disposable and single-use plastic account for the largest share, including packaging for personal care products which often last for only a few weeks but result in long-term damage to the planet. So, why is packaging still made from materials such as plastic, that take more than 400 years to decompose?

Our solution: By turning the product into packaging, we have transformed the concept of packaging-free cosmetics to design a sustainable SOAPBOTTLE for liquid care products. Unlike any other packaging, the SOAPBOTTLE is made from

Below the description is a table with the following data:

Funding Goal	Amount Pledged	End Date
5.53 Eth	0 Eth	6/25/2021

Below the table is a "Donation Amount" field with a value of "1" and a "Donate" button.

Overlaid on the right side of the browser window is a MetaMask Notification window. It shows the transaction details for a "CONTRACT INTERACTION" on the "Rinkeby Test Network". The transaction amount is 1 ETH. The gas fee is 0.000177 ETH, and the total amount is 1.000177 ETH. The window includes "Reject" and "Confirm" buttons.

On the public testnet blocks take around 15 seconds to mine, after this you will receive a confirmation and the values will have updated.