

# Programming Assignment: Crowdfunding an Invention

ELEN E6883: Introduction to Blockchain Technology

March 9, 2020

## 1 Background Description

**Crowdfunding an invention:** Assume that your team invented a wireless charging device for smartphones, called “*MagCharging*”. A MagCharging consists of a plastic case (about \$5) and a copper coil (about \$3), as show in Fig. 1. Now, you want to raise funds on Ethereum blockchain to manufacture 1,000 MagChargings and sell them on Amazon.

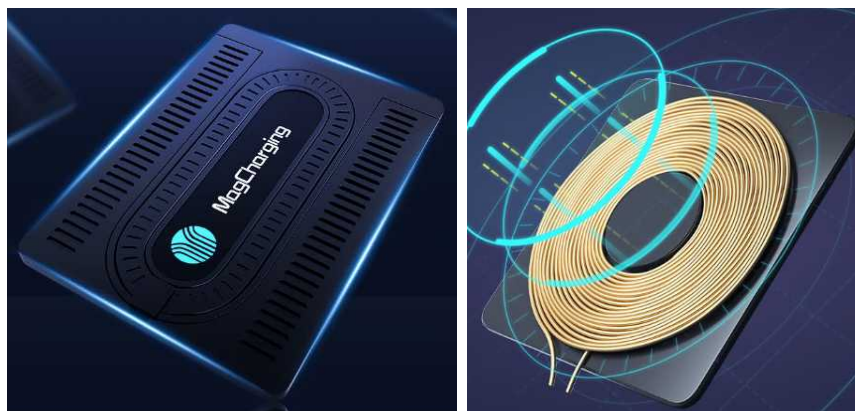


Figure 1: MagCharging: plastic case and copper coil

**Background on Crowdfunding:** A good invention may take a lot of funds and collective efforts. You could ask for investments, however, investors prefer projects that are more likely to get fundings (Series ABCDE... rounds). Therefore, crowdfunding would be an alternative solution.

Crowdfunding is similar to ICOs (Initial Coin Offering). In particular, a startup team first announces an invention and a date for fundraising for that invention. Before the crowdfunding campaign begins, the startup needs to release documents such as whitepaper to introduce the invention. Once the crowdfunding campaign begins, the startup team accepts contributions from investors on the Ethereum blockchain. If the goal (e.g., \$5M) is not met, the funds will be returned to the investors, reducing the risk for investors. If the goal is achieved, the startup keeps the fund and then uses it to develop the proposed product. Since the crowdfunding rules are open and auditable, there is no need for a centralized, trusted platform. The only fee every investor needs to pay is the transaction gas fee on Ethereum.

The major difference between crowdfunding and ICOs is the reward to the investor. In an ICO, participants receive tokens. On the other hand, crowdfunding backers (investors) can get some other type of reward, e.g., a physical product.

In this assignment, you will be asked to deploy a crowdfunding smart contract on the Ethereum and simulate the process of running the proposed MagCharging project (a simplified version of real-life fundraising). You need to submit a report by answering all the questions in this assignment.

## 2 Environment Configurations and Warming Up

### 1): Install MetaMask to your (Chrome) browser

Website: <https://metamask.io/http://personal.stevens.edu/~sdai1/>

MetaMask is a secure identity wallet for Ethereum. It allows you to hold ether & tokens, and serves as your bridge to decentralized applications.

Create two accounts in your MetaMask: **Account 1** and **Account 2** (accounts are wallets that hold ETH, ERC-20s, etc.). We will need multiple accounts in our simulation tasks (Questions 2, 3, and 4).

### 2) Obtain testing Ethers

Post a tweet containing your address (say **Account 1**) on facebook, Twitter or Google+;  
Copy that tweet's link to <https://faucet.rinkeby.io>  
Then, your **Account 1** will receive testing ethers.

Testing: Send one Ether from **Account 1** to **Account 2** (you will use it in Question 4).

### 3) Remix (Solidity IDE)

Website: <https://remix.ethereum.org>

A good tutorial web "Building Smart Contracts with Remix":

[https://remix.readthedocs.io/en/latest/workshop\\_Building\\_smart\\_contracts\\_with\\_Remix.html](https://remix.readthedocs.io/en/latest/workshop_Building_smart_contracts_with_Remix.html)

### 4) Create a crowdsale contract in Ethereum

<https://www.ethereum.org/crowdsale>

### 3 Task Simulations on Smart Contracts

The file “CrowdFunding.txt” is provided as a template to simulate this crowdfunding campaign.

#### Question 1 (Environment settings for our simulations) [15 points]:

Please figure out the proper environments for our simulation tasks in this section. (i) what mode should we select in MetaMask? (ii) what mode should we select in Remix IDE? (iii) how to check the list of all events (i.e., actions to make smart contracts on the blockchain) in our simulation tasks.

#### 1) Complete the file “MagCharging.sol” [10 points]

Open Remix (Solidity IDE): <https://remix.ethereum.org>,

Create a new file “MagCharging.sol”,

Copy codes from “CrowdFunding.txt” into “MagCharging.sol”.

Throughout the sol file you’ll see **TODO tags(total 5)** in the comments. This is where you should insert your own code to make the functions work! If you get stuck, we encourage you to **go through the whole file and get a sense how certain functions work**(you will find hints for your own codes)!

#### 2) Compile the file “MagCharging.sol”

Select a compiler version: 0.4.17+commit.bdeb9e52 (indicated in the first line of the codes),

Compile and you will observe Fig. 2.

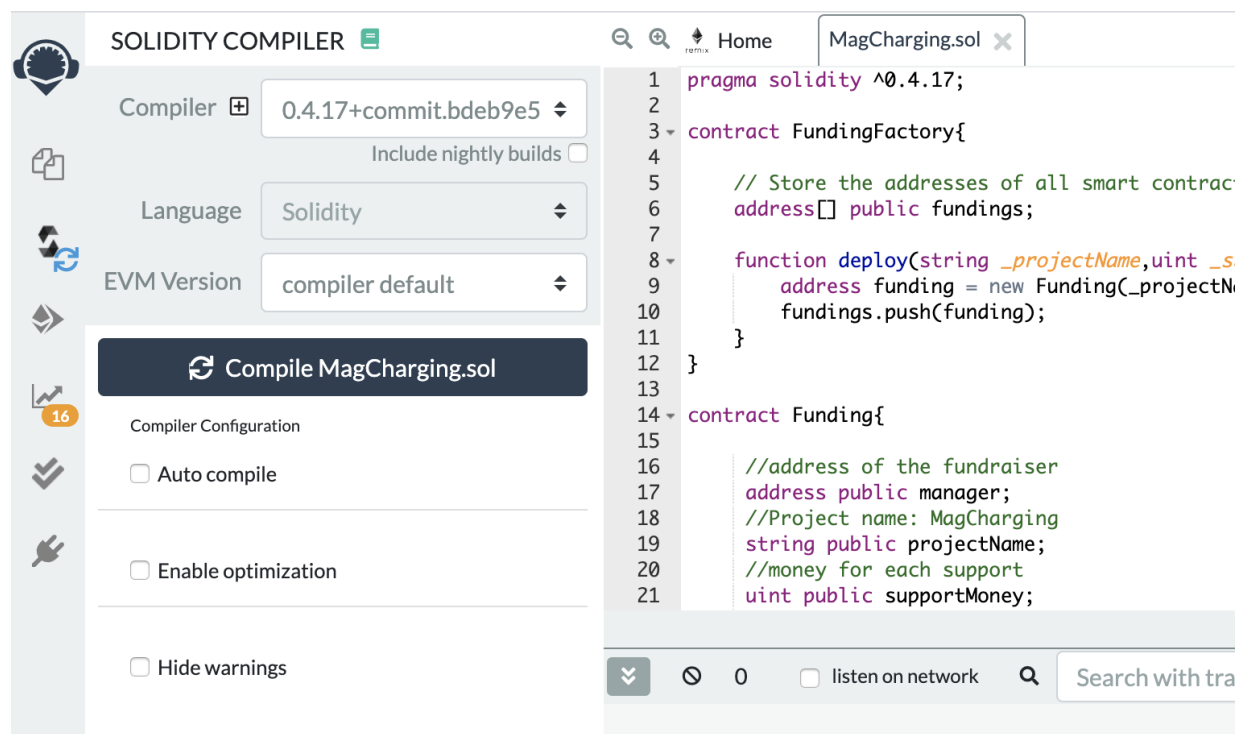


Figure 2: Compile the file: MagCharging.sol

### 3) Deploy the CrowdFunding project on Ethereum blockchain.

Select “Fundingfactory”, click “deploy”, and confirm in MetaMask, then confirm a new deployed contract;

Setup a fundraising by filling: projectName, supportMoney, goalMoney;

Making smart contract: click “transact”, use MetaMask to confirm the deployment, and wait for the Ethereum to add it into a new block. Then, you will observe Fig. 3.

In Funding Factory, we can find the new address for this smart contract. Using this address, we can check the details of this CrowdFunding campaign.

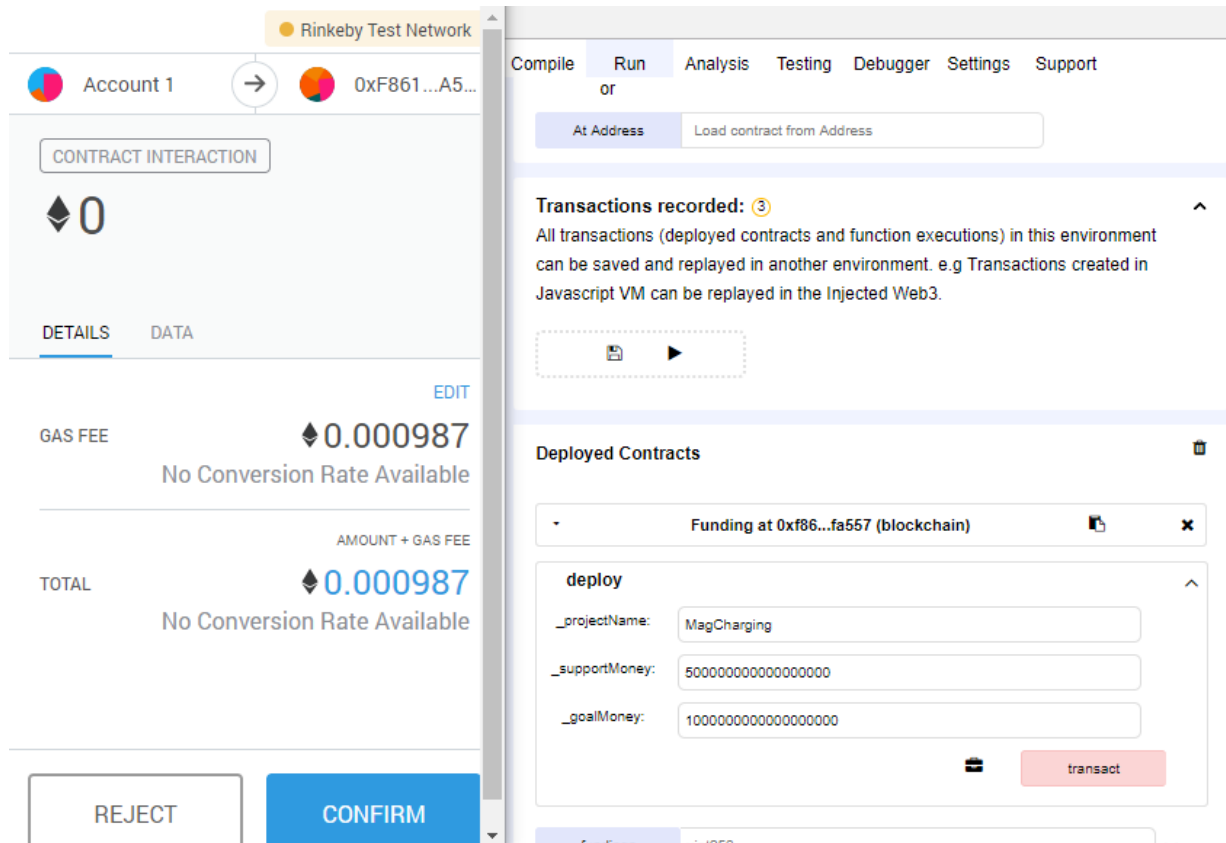


Figure 3: Deploy a contract.

**Question 2 [15 points]:** Please figure out how to invest this CrowdFunding campaign. Attach one or two key screenshots. You can invest multiple times, check the balance, number of investors, etc. For example, please set the fundraising time to be 2 weeks. Assume that  $1 \text{ BTC} = \$1,000$ , please set the goal of the fundraising and each support money properly. Recall that we want to raise money to manufacture 1,000 MagChargings.

**3) As shown in Fig. 1, the MagCharging consists of a plastic case and a copper coil. Please simulate the scenario: your team applies to use money to buy the required materials: plastic case and copper coil.**

**Question 3 [15 points]:** Your team makes two separate requests to buy those two types of materials that will be used to manufacture 1,000 MagChargings. Describe your operations (parameter setting, button click, the procedure of sending out the request on smart contract, etc.) and attach one or two screenshots.

**Question 4 [15 points]:** Then, the investors vote to approve the requests. When more than 50% inventors agree, the request will be approved. Describe your operations (parameter setting, button click, the procedure of sending out the request on smart contract, etc.) and attach one or two screenshots. Hint: for simulation purpose, you can send money to two addresses (representing these two factories).

#### **More questions:**

**Question 5 [20 points]:** Please specify the meanings of the following parameters. Hints: You can check these parameters by simulating actions, check the value changes, then figure out their meanings.

endtime  
getplayers  
getplayerscount  
gettotalbalance  
goalmoney  
requets  
supportmoney  
approvwrequest  
createrequest  
finalizerequest

**Question 6 [10 points]:** Please list at least 5 pros or cons of CrowdFunding via blockchain;.