

A decentralized blogging cum messaging application built on top of ethereum platform

Vignesh Varadarajan , Sohan Sathyan, Kiruthiga Sekar
BLOCKCHAIN AND APPLICATIONS - FINAL PROJECT

Abstract—A decentralized multipurpose idea exchange platform, that enables secure blogging and messaging among the users of the application. Users of the application can create a profile, they can either create a blog or visit other blogs, they can also applaud/like the blog, if they like, which will trigger a transaction over the blockchain network, which will transfer a particular amount of money from the reader to the author of the blog.

I. INTRODUCTION

In today's digital world, internet is biggest medium of mass communication. Ideally people should be able to express themselves freely, but the powerful entities like a government institution or a large corporation often creates hindrance to your ideas from reaching the common mass by either censoring your blog or by deleting your blog completely. In addition to this, currently available platforms pay the bloggers for the advertisements that are displayed in their blogs and not for the blog itself.

We are tackling these problems by building a decentralized blog application using smart contracts that runs on top of the Ethereum blockchain. The main features of our application are: (1) It is decentralized, (2) It provides a way to incentivize the author without having any advertisements and (3) Provides a messaging system which eases the communication between the author and the readers of the blog.

The decentralized nature of the application ensures that there is no central authority or a powerful entity to control the activities on the application. As the application runs on Ethereum blockchain, we observe that the system is resistant to censorship such that once an author has the control over his blog.

This application allows the readers to incentivize the author if they enjoy reading the article. Whenever a article is applauded, a new transaction is initiated, which will transfer bitcoins from the reader to the author. This will avoid the advertisements, that are totally irrelevant to the blogs.

This application also provides a messaging system wherein any two users can communicate with each other to exchange their ideas in a secure environment. This messaging system would help the readers to get more insights about the article by communicating with the author. This messaging system can be used by any users of the platform to communicate with one another.

The sole intention of this project is to develop a secure environment for people to express their thoughts with no

government or any other organization controlling the activities of the bloggers and censoring any articles, based on a criticized topic.

II. RELATED WORKS

a. Medium Blogs :

Medium is an online platform for publishing blogs that supports clapping to encourage a blog. A blog can be clapped up to 50 times, which potentially describes the quality and the likeliness of the post.

b. Steemit :

Steemit is a blockchain based blogging platform that encourages people for publishing high quality content by rewarding them with cryptocurrencies. This concept is similar to Medium, but it uses the blockchain for the reward system. As people upvote any post, the authors are rewarded with STEEM Dollars, which can later be traded for Bitcoins or Ethereum.

c. Whatsapp :

WhatsApp Messenger is a freeware and cross-platform messaging and Voice Over IP (VoIP) service owned by Facebook. WhatsApp's messaging service ensures that each message is encrypted and only the users who belong to the conversation can decrypt the messages and read them.

III. IMPLEMENTATION

We have developed a decentralized platform for blogging cum messaging that allows a user to create a blog to talk about any topic of their interest without having the fear of being censored. The decentralized nature of our application means that no single entity has control over the platform. The application also provides a medium to communicate with other users. When a reader likes a blog, they can applaud/like the blog, which will trigger a transaction between the reader and the author of the blog.

A. Technologies used

a) *Solidity*: Solidity is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behaviour of accounts within the Ethereum state. We have used Solidity version 0.5.0 in our project.

b) *Truffle*: Truffle is a development environment, testing framework and asset pipeline for Ethereum, aiming to make life as an Ethereum developer easier. We have used truffle to compile and deploy our smart contracts. We have used Truffle version 5.0.7 in our project.

c) *Ganache*: Ganache is a personal blockchain for Ethereum development that can be used to deploy contracts, develop your applications, and run tests. We have used Ganache version 2.0.0 in our project.

d) *Metamask*: A google chrome extensions that allows us to run Ethereum dApps right in the chrome browser without running a full Ethereum node. We have used Metamask version 6.4.1 in our project.

e) *web3.js*: The web3.js library is a collection of modules which contain specific functionality for the Ethereum ecosystem. Among the collections, we use

- 1) The web3-eth is for the Ethereum blockchain and smart contracts
- 2) The web3-utils contains useful helper functions for DApp developers.

B. Detailed Implementation

a) *Blog*: For blogs, we have used a mapping to store the generated ID of the blog which the key and its contents as the value which is the blog itself. Each Blog itself is saved as a struct, a data structure in solidity that allows us to group all properties of a particular object. A mapping is used to structure value types, such as booleans, integers, addresses, and structs. It consists of two main parts: Key and a Value

b) *Chat*: The blockchain is a list of blocks which are fundamentally lists of transactions. Each transaction has an attached receipt which contains zero or more log entries. Log entries represent the result of events having fired from a smart contract. In the Solidity source code, to define an event, you mark it thus by preceding it with the event keyword (similar in usage to the function keyword). You then call or fire the event in the body of whatever function you wish to cause to generate the event. You may fire events from any function using the emit keyword. Now, using web3.js which provides functions to filter and retrieve appropriate logs and fetch the chats from there

c) *Profiles*: Each profile is saved as a struct. Here we save the first name, last name as part of the struct. Each profile is uniquely identified by their address. This relationship is provided using maps in solidity.

d) *Applaud*: This feature provides a way to transfer ethers between two accounts. When an author selects a particular blog and chooses to applaud it, the sender and receiver accounts are captured and 1 ether is transferred from sender to receiver.

e) : All the transactions that updates the blockchain would require a gas as part of each transaction. Gas is a unit of measuring the computational work of running transactions or smart contracts in the Ethereum network. Hence, the accounts that make any of the transactions must have sufficient balance in their account.

save the blog data and the message data(message data is saved in the logs). We have come to a realization that as the price of ethereum increases, the cost to write a blog and to send messages also significantly increases for the end user. This would make the application less attractive for the end user to use. One more thing that surprised us was that anyone in the ethereum blockchain can query the logs. As part of the future work of the project, we would like to do two things:

- 1) Explore the possibility of storing the blog data outside blockchain.
- 2) Encrypt the message when a user is sending it and decrypt the message when he or she wants to retrieve it.

REFERENCES

- [1] <https://solidity.readthedocs.io>
- [2] <http://www.dappuniversity.com/>

IV. CONCLUSIONS

In this project we have described an approach to build a decentralized blogging cum messaging application on ethereum platform. In our approach, we have used the blockchain to