Decentralized Application

Gokulahari.U , Padmavathy.C , Aishwarya.PL, Dhanuu Shri. R.V

Department of Computer Science, Engineering, Sri Ramakrishna Engineering College
Vattamalaipalayam, NGGO Colony post, Coimbatore

Abstract *— Since people need multiple applications to be run in a particular organization, DApps comes into use. Network coverage in some organizations might be incapable of carrying much load when asked to download the application every time.*

Instead, the DApps contain multiple applications in which any individual can install the desired application and make use of it. Some of the general characteristics of DApps are: Open source, decentralized, incentive and they have an algorithm. If the Source code of the app is available to all, then it is called Open source. This DApp provides time to time installation depending upon the user requirement.

Keywords *— Decentralized Application, Car Pooling, Ethereum, Sustainable Mobility.*

I. INTRODUCTION

A decentralized application is a computer application that runs on a distributed computing system. DApps have been popularized by distributed ledger technologies such as the Ethereum Blockchain, where DApps are often referred to as smart contracts. DApps is an acronym for remembering the five qualities for effective goals of Dated-specific deadline for achievement, Achievable-Goals must be realistic, Personal- Goals must be what the user needs for themselves and not what others want for them, Positive-Goals must focus on what we are intended to rather than what we are not, Specific-Goals must be stated in specific and not in measurable terms.

A. Key findings of DApps

- The popularity of DApps -The distributions of the number of users, transactions, and transaction volumes against the popularity of DApps typically follow the Pareto principle, i.e., a few DApps have substantial popularity. DApps with financial features has a large influence on the market.

- Growth of DApps -The number of DApps was first rapidly grown for the categories including Exchanges, Gambling, and Finance.

As more DApps are developed, the number of DApps from the high-risk category increases significantly, leading to potential security issues.

- Open source of DApps -Currently, the open-source levels of DApps are a bit diverse and not satisfactory. Only 15.7% of DApps are the fully open source where the code of both the DApp and related smart contracts are available.

- In contrast, 25.0% of DApps are completely closed source. In general, DApps whose smart contracts are open source, usually has more transactions than others, indicating that open source can have potentially significant impacts on the popularity of DApps.

- Usage patterns of smart contracts-Asmart contract is a computer program or a transaction protocol which is intended to automatically execute, control or document legally relevant events and actions according to the terms of a contract or an agreement. 75% of DApps consist of only one smart contract. For DApps with multiple smart contracts, there are 3 usage patterns of smart contracts: leader-member where smart contracts are invoked with each other through internal transactions, equivalent where there is no invocation between smart contracts, and factory where the child contracts are deployed by a factory contract.

- Cost of deploying smart contracts -The average deployment cost per smart contract for single-contract DApps is less than that for multi-contract DApps. The deployment costper smart contract is correlated with the line of code (LoC)and the number of functions(NoF)where NoF has more influence on the deployment cost.

- Cost of executing smart contracts –In the median case, only 50% of the prepaid fee for executing smart contracts isused, leading to half of the prepaid fee being locked in transactions until they are confirmed.
- End-users can choose suitable DApps according to DApp distribution; end-users and developers can

set the proper amount of prepaid fee for executing smart contracts to avoid their assets being locked during contract executions;

- DApp developers can understand the DApp life cycle better and use a suitable pattern to design the DApp architecture and smart contracts; blockchain vendors can optimize some mechanisms to serve DApps better.

The above mentioned are some of the notable key findings of DApps. Today the role of Decentralized application has become very important in all the emerging technologies.

II. RELATEDWORK

The existing system describes the features of the previous working model and its drawback. A descriptive analysis of the popularity of DApps, summarize the patterns of how DApps use smart contracts to access the underlying blockchain and explore the worth-addressing issues of deploying and operating DApps. Based on the findings, some implications for DApp have been proposed. DApp developers have to improve the efficiency of DApps, and for blockchain, vendorstoen hance the support of DApps The advancement of Information and Communication technology has brought a lot of changes in all spheres of the daily life of human being. E-commerce has a lot of benefits that add value to customer's satisfaction in terms of customer convenience in any place and enables the company to gain a more competitive advantage over the othercompetitors.

This study predicts some challenges in an emerging economy. In some countries, due to the huge student population, college buses were not provided. Instead, the concept of carpooling comes into existence. The largest DApp market has been chosen, Ethereum DApps, to conduct the empirical study. In this section, some background knowledge of the Ethereum Dapps is given.

. Ethereum DApp Ethereum blockchain provides computation and storage capabilities via the mechanism of smart contracts. Therefore, Ethereum DApp scan deploysmart contracts to use the capabilities provided by Ethereum to implement business logic. In theory, all the processes and data of a blockchain-based DApp should behandled and stored on the blockchain for pure decentralization. However, due to the performance bottleneck of state-of-the-art blockchain systems, current DApps usually implement only parts of their functionality on the blockchain. Asaresult, three kinds of architectures are adopted by Ethereum DApps in practiceas

shown in Figure 2: direct, indirect, and mixed. For DApps of the direct architecture (Figure 2a), the client directly interacts with smart contracts deployed on the Ethereum. DApps of the indirect architecture (Figure 2b) have back- end services running on a centralized server, and the client interacts with smart contracts through the server. DApps of the mixed architecture combines the preceding two architectures where the client interacts with smart contracts both directly and indirectly through a backend server.

III. DESIGN CONSIDERATIONFOR DAPP DEVELOPERS

Modern web applications are based on an infrastructure in which a single point of failure naturally exists. DApps aims to alleviate these issues by distributing critical components that store data or parts of infrastructure between various peers or nodes. That's why, when designing DApp, security, cost, usability features should be taken into consideration. DApp applications are required to contain some features. Such as;

- Better performance (low latency, high throughput),
- Reasonable low transaction fee,
- Flexible maintainability, DApps should not store or replicate user data.

The point of Blockchain security is based on not having any central vulnerability. Users should use the application as the sole administrator of their independent and unique identifiers. It reduces reliance on central authorities. The application should not be too complicated and should have a simple interface. Unnecessary coding should be avoided; it increases the cost and may create a security weakness. It is noted by some researchers that the use of blockchain technology is not always useful in preparing a secure software system. Developers should first learn the basic concepts of blockchain and current blockchain technologies and applications. It is stated that it would be more appropriate to start working on application development.

In our context, a DApp is a "blockchain-enabled" web application running on a peer-to-peer computer network, rather than a single server. It includes both front end and back end and works independently on all nodes. Typically, other applications also use the same technology to create the front-end interface. The only critical difference is a smart contract that connects the application with a blockchain network.

A smart contract is a self-executing contract that contains the conditions for writing the contract between the buyer and the seller directly in line with computer codes. Smart contracts allow reliable transactions and agreements to be made between different, anonymous parties, without the need for a central authority, a legal system, or an external implementation mechanism.

IV. PROPOSEDSYSTEM

A. Car Pooling

Carpooling is the sharing of car journeys in which more than one person travels to the same destination to reduce each person's travel cost. Drivers and passengers often search for their journey through one of the several mediums available. Once they find a match, they contact each other to arrange the details for the journey. It is not necessary that every passenger has to travel throughout the journey with the other person. Benefits of this system are:

- The prolonged life cycle of a vehicle.
- Save time and reducestress.
- Reduce Traffic.
- Maximize the commute time.
- Reduce carbon footprint.
- Use carpool or High-Occupancy Vehicle(HOV)

We have taken carpooling as one among the apps in Dapps and used Python anywhere as the server along with Firebase as the database.

B. Online Grocery Store

An online grocery store allows online ordering or a standalone e-commerce service that includes grocery items. There is usually a delivery charge for this service. Online grocery delivery services are available throughout Europe, Asia, and North America, mostly in urban centers. The online ordering is done through e-commerce websites or apps. Some of the features of Online grocery store are as follows:

- Saves Time
- Saves Fuel Cost
- Convenient to use

Online Grocery Shopping is another app in Dapps and we have used Microsoft Azure as the server with firebase as the database.

V. WORKINGPRINCIPLE

The project study focusses to implement Dapps in which it contains multiple applications in it. Any individual can install the app that is needed for them in their dedicated organization. The working principle of Dapps is that it consists of multiple apps and each app will be connected to a separate server and all the servers will be connected to a single database. In case, of any server has a problem, the app connected to that server doesn't stop working. It gets boosted up by another server, which is connected to that common database. Here comes the concept of Dapps, If one server comes down, it uses other servers and gets boosted up. Though they appear as a centralized one, they work as a decentralized application.

VI. DESIRABLE CHARACTERS OF DAPPS

DAPPS According to the application scenarios discussed above, future dApps demand a blockchain platform that fulfills the following desirable characteristics:

A. BETTER PERFORMANCE

- Low Latency

Long transaction delay has been a critical issue since the birth of Bitcoin. Since the average time for the Bitcoin nodes to mine a block is 10 minutes, the average transaction confirmation time is around an hour (as a user typically waits for 6 blocks). Even though the response latency has been significantly reduced to around 15 seconds in Ethereum, a sufficiently small latency to support interactions of general applications is yet to be achieved. Longer delays frustrate users and make dApps less competitive with existing non-blockchain alternatives. For instance, a common user in a blockchain-based social network website will typically require the system to respond to his/her like or share action to a post within 2 to 3seconds.

- High Throughput

Modern web-based systems, e.g., social

networks, massive multi-player online games, online shopping malls, require the blockchain platform to support millions of active users daily. Therefore, the capability of handling a large amount of concurrent traffic is critical in a DApp platform. However, current blockchain platforms still suffer from throughput bottlenecks.

For example, Crypto Kitties, which gained a lot of popularity on its launch, at one point account for nearly 30% of all transactions on Ethereum, which resulted in a peak backlog of about 30,000 pending transactions.

- Fast Sequential Performance

In system designs, dependencies among software components or logical steps restrict the execution of an application. Some procedures in certain applications, such as updates on one particular piece of data, cannot be implemented in parallel, due to the sequential dependent on the results produced by previous steps. In blockchain systems, the sequential performance of a DApp is determined by the response delays from all nodes in the network, since all transactions/operations should be executed and verified by all nodes to reach a consensus. Therefore, the blockchain platform that hosts DApps needs fast sequential performance to handle high volumes.

B. ENABLING OFFLINE TRANSA- CTIONS

Many current blockchain systems depend on Internet connectivity to verify funds quickly. Systems participating in a particular blockchain network may go offline periodically.

However, if a subset of devices disconnects from the Internet and exchange signed transactions with each other, there is no guarantee that double spending has not occurred if another device remaining online with the same key-pair as an offline device can simultaneously spend. For example, consider a group of people takea bustrip to a decentralized Applications: The Blockchain-Empowered Software System remote village with their mobile phones. The village has no Internet access. A DApp could be designed such that it could accept offline transactions that are signed for payment of goods. A person on the bus could send their payment for coconut this way to a vendor using a local Bluetooth connection. When this signature eventually is relayed to the Internet at a later time, the payment would be successful, unless the person on the bus also had the same key- pair being used back in their home computer, and spent the money before they went offline. This problem becomes more complicated when large groups of devices fragment the network. Since many of the blockchains rely on over 51% of devices to cooperate, there are potential malicious attacks possible whereby an attacker could attack the Internet infrastructure strategically to divide and conquer with 51%attacks.

C. REASONABLE MONETARYCOST

- Low Transaction Fee

As part of the incentives for block producers, the concept of transaction fee was born with Bitcoin. In classic blockchain systems, e.g., Ethereum, transaction fees can also be a way to prevent spams or malicious executions of 0.smart contracts, since intruders need to spend their tokens to start their attacks. However, transaction fees become a barrier for transactions with relatively small monetary values, due to the large proportion of the transaction overheads. In the current blockchain ecosystem, the DApp developers are struggling with the high transaction fees they need to pay when they deploy and execute their smart contracts.

□ Modern Free Internet Business Model

Another critical issue related to transaction fees is

the business model. By default, the action initiator, e.g., the invoker of the smart contract in Ethereum, need to purchase tokens before they can utilize the system. This limits the user base of the DApp, especially since crypto currency has yet to achieve universal acceptance in society. The modern Internet business model is based on the fast increase of user popularity, which implies that the DApp developers should have the flexibility to offer users free services. In other words, the users do not need to purchase or hold tokens to use the platform, which leads to more widespread adoption. Future DApp can adopt the modern Internet business model by offering free services to users and share the profit of the platform with its users and its content producers

D. FLEXIBLE MAINTAINABILITY

- Enabling System Upgrades

As blockchain technologies are still in their infancy, a blockchain system will inevitably require upgrades from one version to the next. However, due to the nature of P2P consensus, the hard fork is the only approach for current blockchain systems to upgrade them selves, which may result in the loss of participating network nodes. Another potential issue for a hard fork is that there will be multiple similar tokens sharing a common origin, which will confuse users. For example, Bitcoin and Bitcoin Cash, 'Ethereum' (ETH),and' Ethereum Classic'(ETC)forked from each other in July 2016. To this end, a system upgrade mechanism is needed fornext-generation blockchain systems, which facilitate version control of DApps deployed overthem.

- Easy Bug Recovery

Security issues in smart contracts have been investigated in many previous works]. Though most bugs and system flaws can be prevented by careful implementation and intensive tests, it is virtually impossible to guarantee that anon-trivial smart contract is bug-free. The situation is exacerbated by the high complexity of some DApps.

VII. SYSTEM SETUP AND IMPLEMENTATION

The first dependency we need to do is to install Python and set up the Django Framework.Install Python 3.6. To determine the installed Python version, open the

Terminal application, type the following, and press Return:**python–V**.Setthe Environmental variablesforpython3.6andpython3.6//Scripts. Install virtualenv, using the following command to prevent different versions of libraries/packages from messing with each other. **pip install virtualenv**.Create a new folder using the cold, **mkdirmyproject** **cdmyproject**.Inside myproject folder use following command.**virtualenv venv.** Activate virtual environment **venv\Scripts\activate** command**.** Install Django 1.11.4 inside the virtual environment using **pipnstall django==1.11.4.**

Start a new Django project, run the python Django production server using **“python manage.py runserver 8000”**, this is done to ensure whether Django is working or not. To create the first app,go to the directory where the manage.py file is and executes the following command: **Django-admin startapp boards**

Configure the created app in settings.py. Python functionalities should be written in views.py. Totell Django when to serve this view, and is done inside the urls.py file. Run the python Django production server by opening the URLhttp://127.0.0.1:8000.

To create a project in firebase, redirect to https://console.firebase.google.com.

After creating the project, initialize the firebase by adding the Firebase SDK's to the web app.

Add a defer flag to each script tag for the Firebase SDKs, then defer the initialization of Firebase using a secondscript.

To deploy the developed application, create a beginner account at Python Anywhere. Go to console and type the following **$pwd**. Next, clone the project from Git-repository. It is better to clone the repository than uploading the

https://carpooling.pythonanywhere.com.Now, create the virtual environment at pythonany where, and activateit. $virtualenv env -p python3.6

$source env/bin/activate

Go to Web and click on Add a new web app Select a Python Web framework: Manual configuration and select python version: 3.6. Click on the link for WSGI Configuration. Uncomment the lines which are related to the Django Project. Provide the location of the virtualenv. Press the Reload button, whenever we change anything in our project. Now, basic settings are completed and the webpage can beseen.

VIII. RESULT

A new car-pooling system is proposed. The objective is to develop an intelligent and reliable transportation solution that reduces travel costs, traffic, and parking conditions. Preliminary tests have been performed to access the efficiency of the model in different scenarios. The simulation results based on real data show that the model provides a high-quality solution in a reasonable amount of time.

TABLE I.TEST RESULT FOR CAR-POOLING

Metrics	No Preferences	With Preferences
Min/max/avg increase in distance(%)	0/20.2/4.8	0/28.6/7.2
Late arrivals	0	0
Used Cars	35	32
Age Group(%)	26.05	16.8
Car Capacity(%)	11.43	18.75

Few Apps run backend, but most of the apps don't whereas Dapps run mostly backend and so if any flaw occurs in DApp the data is retained. Incase of Apps the data will be lost.

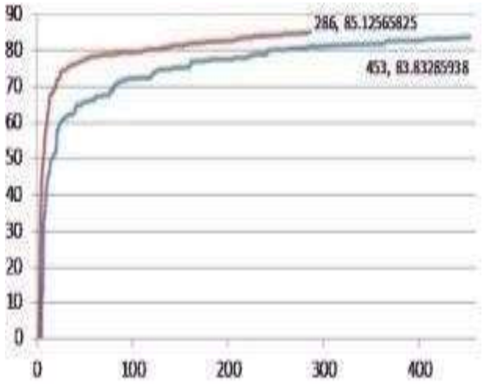

Fig. I APP Vs DAPP

Fig 1 depicts the performance of Apps and DApps. The red line in the graph shows the performance of the normal App whereas the blue line shows the performance of Dapps.

IX. CONCLUSION

In this present era, we all need many mobile applications. Rather than searching and installing each application, we can install a single app called Dapps. It contains multiple apps, in which the user can install the app needed for him. If any individual installs a single app and if its server gets down, then he will not be able to continue his work in that app. Where as, this problem gets solved in Dapps. Here, if any server gets down, the app connected to that server gets boosted up by the server connected to that common database.

ACKNOWLEDGMENT

A Successful project is a dreamcome true for any student and to fulfill this dream, efficient teamwork is required. I am thankful to a lot of people who are directly and indirectly involved in this project. I am thankful to my guide Mrs.C. Padmavathy Assistant Professor, Department of Computer Science and Engineering for providing the necessary facilities for carrying out this work successfully and for the helpful tips and timely suggestion, without whose assistance, I would have faltered in this effort.

REFERENCE

[1] "*Electronic Commerce: A Study on Benefits and Challenges in an Emerging Economy*" By Abdul Gaffar Khan lana Bhashani Science and Technology University, Bangladesh.

[2] "*Smart Peer Car Pooling System"* By Syed Imran, Department of Computing Middle East College Muscat, Sultanate of Oman,2016 3rd MEC International Conference on Big Data and SmartCity.

[3] "*An Intelligent Carpooling App for a Green Social Solution to Traffic and Parking Congestions*" Oussama Dakroub, Carl Michael Boukhater, Fayez Lahoud, Mariette Awad, Hassan Artail Department of Electrical and Computer Engineering American University of Beirut Beirut,Lebanon.

[4] "*A First Look at Blockchain-based Decentralized A Applications*" By Kaidong Wu, Yun Ma, Gang Huang, Xuanzhe Liu Key Lab ofHigh-Confidence.

[5] Amaldev P J, Gokulkrishna S, Mohammed Salman, Prof. Elizabeth Issac "*Decentralising Stock Exchange Using Blockchain*" International Journal of Computer Trends and Technology 68.5 (2020)

[6] "*Real-Time Carpooling System*" N. V. Pukhovskiy, R. E. LepshokovØstfold University College; N-1757 Halden;Norway.

[7] "*Dapp.com 2018 Dapp Market Report*," https://www.dapp.com/article/ annual-dapp- market- report-2018, January2019.

[8] "*DApp Survey Results 2019*," https://medium.com/fluence-network/ dapp- survey- results-2019-a04373db6452, January 2019.

[9] "*What is dApp? Decentralized Application on the Blockchain,* "https://blockchainhub.net/decentralized- applications- dapps/, 2019.

[10] Sandeep Kumar, Abhay Kumar, Vanita Verma "*A Survey Paper on Blockchain Technology, Challenges and Opportunities*" International Journal of Computer Trends and Technology 67.4 (2019)