ELECTRICITY CONSUMPTION AND BILLING DATA EXCHANGE SYSTEM IN BLOCKCHAIN ENVIRONMENT

A PROJECT REPORT

Submitted to

ASSAM DON BOSCO UNIVERSITY

by

FLORIGINIA SHADAP	**-**	**DC2019BTE0121**
SIMANTA RAJBANGSHI	**-**	**DC2018BTE0106**

in partial fulfilment for completion of Mini Project of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCHOOL OF TECHNOLOGY ASSAM DON BOSCO UNIVERSITY

AZARA, GUWAHATI 781 017, ASSAM, INDIA.

BATCH (2018- 2022)

CERTIFICATE

This is to certify that the Project Report entitled "**ELECTRICITY CONSUMPTION AND BILLING DATA EXCHANGE SYSTEM IN BLOCKCHAIN ENVIRONMENT"** submitted by **Floriginia Shadap (DC2019BTE0121), Simanta Rajbangshi (DC2018BTE0106)** to Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for Minor project of 5th semester of Bachelor of Technology. It is a bonafide record of the project work carried out by them under my supervision during the year 2020.

(Signature of the Internal Guide)

Mr Kausthav Pratim Kalita
Assistant Professor
Department of CSE
School of Technology
Assam Don Bosco University

Date: ....

CERTIFICATE

This is to certify that the Project Report entitled "**ELECTRICITY CONSUMPTION AND BILLING DATA EXCHANGE SYSTEM IN BLOCKCHAIN ENVIRONMENT**" submitted by **Floriginia Shadap (DC2019BTE0121), Simanta Rajbangshi (DC2018BTE0106)** to Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for Minor project of 5th semester of Bachelor of Technology. It is a bonafide record of the project work carried out by them under my supervision during the year 2020.

Dr. Bobby Sharma
Head of the Department,
Dept. of Computer Science & Engineering,
School of Technology

Date: ....

Prof. Manoranjan Kalita
Director,
School of Technology

Date: ....

EXAMINATION CERTIFICATE

This is to certify that **Floriginia Shadap (DC2019BTE0121), Simanta Rajbangshi (DC2018BTE0106)** of the Department of Computer Science & Engineering has carried out the Project Work in a manner satisfactory to warrant its acceptance and also defended it `successfully.
I wish **him (/her)** all the success in **his (/her)** future endeavours.

Examiners:

1. External Examiner:

2. Internal Examiner:

3. Internal Examiner:

DECLARATION

I hereby declare that the project work entitled "**ELECTRICITY CONSUMPTION AND BILLING DATA EXCHANGE SYSTEM IN BLOCKCHAIN ENVIRONMENT**" submitted to the Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for Minor project of 5th semester of Bachelor of Technology. It is an original work done by me under the guidance of name of **Mr Kausthav Pratim Kalita (** *Assistant Professor, Department of CSE, School of Technology, Assam Don Bosco University)* and has not been submitted for the award of any degree.

(Signature of the student)

FLORIGINIA SHADAP (DC2019BTE0121)

SIMANTA RAJBANGSHI (DC2018BTE0106)

Department of Computer Science & Engineering
School of Technology
Assam Don Bosco University

ACKNOWLEDGEMENT

We take this opportunity to express our profound gratitude and deep regards to our guide (**Mr Kausthav Pratim Kalita**) for his exemplary guidance, monitoring and constant encouragement throughout the course of this mini project. The help and guidance given by him time to time shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Assistant Professor **Mr. Alexy Bhowmick** and **Mr. Rupesh Mandal**, Project Coordinators, for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We also thank **Dr.Bobby Sharma,** Head, Computer Science and Engineering Department, and School of Technology, Assam Don Bosco University for giving us this opportunity to do our project to explore and increase our knowledge in the field of Computer Science.

ABSTRACT

This project is based on the billing system of Electricity Boards. Electricity bills are generated and submitted to consumers in monthly basis. It is occasionally observed that the billing amounts generated by companies are not satisfactory to the consumers, which includes overpriced or underpriced bills depending on different situations and circumstances. Thus, it is necessary to have a fully transparent mechanism to inspect the bills against the electricity consumption. Lastly, the bills need to paid on monthly basis in the present system. The solution should be able to manage payments on a monthly basis, quarterly, semi-annually, annually or as required by the company and consumer agreement. Thus, a provision should be kept to allow the customers to pay their bills in segments rather than as a whole amount, and this specific provision is usually targeted towards the rural areas of the country, meanwhile the transparency would help reducing corruptions in the billing system to an extent. Also, the implementation of blockchain indicates that the transaction history is completely secure and up-to date and there can’t be any issues related to consumers shown as “not paid” even after the payment.

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

GUI: Graphical User Interface

Geth: Go Ethereum

IDE: Integrated Development Environment

KLOC: Kilo Lines of Code

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Title: Electricity Consumption and Billing Data Exchange System in Blockchain Environment

1.2 Overview:

A blockchain is, in the simplest of terms, a time-stamped series of immutable records of data that is managed by a cluster of computers not owned by any single entity. Each of these blocks of data is secured and bound to each other using cryptographic principles (i.e., chain). The blockchain network has no central authority, it is the very definition of a democratized system. Since it is a shared and immutable ledger, the information in it is open for anyone and everyone to see. Hence, anything that is built on the blockchain is by its very nature transparent and everyone involved is accountable for their actions. This system can be implemented in billing systems as well due to the transparency. It is occasionally observed that the billing amounts generated by companies are not satisfactory to the consumers, which includes overpriced or underpriced bills depending on different situations and circumstances. Thus, it is necessary to have a fully transparent mechanism to inspect the bills against the electricity consumption.

1.3 Objective:

The objective of this project is to create a Billing System based on blockchain for payment of electricity bills. It is a GUI based application to carry out bill payments, check electricity consumption, etc.

1.4 Existing System: Following are some of the existing systems using blockchain technology:

1.Blockchain Based Metering and Billing System Proposal with Privacy Protection for the Electric Network : This system is designed to use blockchain technology and Internet of Things to meter the billing of the customers for electric network. It is aimed towards providing transparency by solving privacy and trust issues in the current system. The IoT device, Raspberry Pi will be used to simulate metering meanwhile the blockchain system selected for the system called Hyperledger Fabric.

2. Blockchain Utilization in Healthcare: Key Requirements and Challenges : This project is focused on exploiting the advantages of blockchain technology in medical and healthcare area, by implementing the same for medical bill payments like operation charges, medicine expenses, bed charges, etc.

3. Blockchain-Based Intelligent Network Management for 5G and Beyond : The current architecture of mobile network is mostly based on centralized management entities which is not suitable for the spectrum and billing procedure. Also, the mobile network slows down the development due to maintenance of balance between capital expenses and profit margins. This system is aimed towards building an intelligent network architecture using blockchain to handle the relation between operators and users by smart contracts.

4. Mobile charger billing system using lightweight Blockchain : Transportation using electricity is emerging very fast. Electric vehicles that require charging at stations require bills to be maintained. A billing system is thus necessary for the operation. This paper proposed a mobile charger billing system that uses Blockchain technology for more secure transactions and better data management as well.

1.5 Proposed Plan:

- A private Ethereum network will be created using Geth.
- A Smart Contract will be written for the system.
- Deployment of Smart Contract will be carried using modules imported from the web3.py library.
- A Graphical User Interface will be developed for the system using Tkinter.

CHAPTER 2
REQUIREMENT ANALYSIS

2.1 SYSTEM REQUIREMENTS:

2.1.1 Hardware Requirements:

SL NO.	TYPE	REQUIREMENT
1	CPU	Intel Core i3(7th Generation) or higher / AMD A10
2	GPU	Intel UHD 520/NVIDIA GT 900 equivalent or higher
3	RAM	4GB preferred

2.1.2 Software Requirements:

SL NO	NAME	VERSION	DESCRIPTION
1	Geth (Go Ethereum):	1.10.1	Geth is an implementation of an Ethereum node in the Go programming language. Geth is a program which serves as a node for the Ethereum blockchain, and via which a user can mine Ether and create software which runs on the EVM – the Ethereum Virtual Machine.
2	Visual Studio Code	1.54.2	IDE used for creation of nodes, using geth, writing programs for interaction with smart contracts, and other coding tasks.
3	Python 3	Python-3	Python is an interpreted, high-level and general-purpose programming language. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

4	Web3.py	5.17	Web3.py is a Python library for interacting with Ethereum. It is commonly found in decentralized apps (dapps) to help with sending transactions, interacting with smart contracts, reading block data, and a variety of other use cases.
5	Operating System	Windows 10	We will be using Windows 10 for this project because of the ease of installation of various packages, IDE’s and libraries.

2.2 FEASIBILTY STUDY:

Feasibility Study is a study to evaluate feasibility of proposed project or system. It is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyze whether software product will be right in terms of development, implantation, contribution of project to the organization etc.

2.2.1 Economic Feasibility: Economic feasibility determines how cost efficient our project is. As already mentioned in the requirements section, there are no economic resources required to build the entire system (excluding the machine used for building), and also to run the final product, no resources are required apart from a working Windows machine. Moreover, the libraries and softwares that are mentioned are free to use. Thus, our project is economically feasible.

2.2.1.1 COCOMO Model: The Constructive Cost Model (COCOMO) is a procedural software cost estimation model. The model parameters are derived from fitting a regression formula using data from historical projects.

The basic COCOMO equations take the form:

Effort Applied (E) = a(KLOC)b [person-months]

Development Time (D) = c(Effort Applied)d[months]

People Required (P) = Effort Applied/Development Time [Count]

where, KLOC is the estimated number of delivered lines (expressed in thousands) of code.

The coefficients a, b, c and d are given in the following table:

Software Project	a	B	c	D
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.2	2.5	0.32

Table 1: COCOMO coefficient values for different project types

Our project type is **Organic**

Estimated LOC = 1000

Now basic COCOMO equation of our project is:

Effort Applied (E) = a (KLOC)^b person/month

= 2.4(1)$^{1.05}$ person-month

= 2.51 person-month

Development Time (D) = c (Effort Applied) d [months]

= 2.5 (2.51) 0.38 [months]

= 2.375 months

= 2.4 months (approximately)

People Required (P) = Effort Applied/Development Time [count]

= 2.51/2.4 [count]

= 1.04 [count]

= 2 (approximately)

2.2.2 Scheduled Feasibility: Scheduling the project task is an important project planning activity. We have scheduled our project based on the estimated hours that will be required and also on the total days that are available to work. It is found to be feasible in the allotted time slot and it is illustrated with the help of Work Breakdown Structure and Gantt chart provided in the next pages.

2.2.2.1 Work Breakdown Structure: This is the Work Breakdown Structure for our project: Electricity Consumption and Billing Data Exchange System in Blockchain Environment.

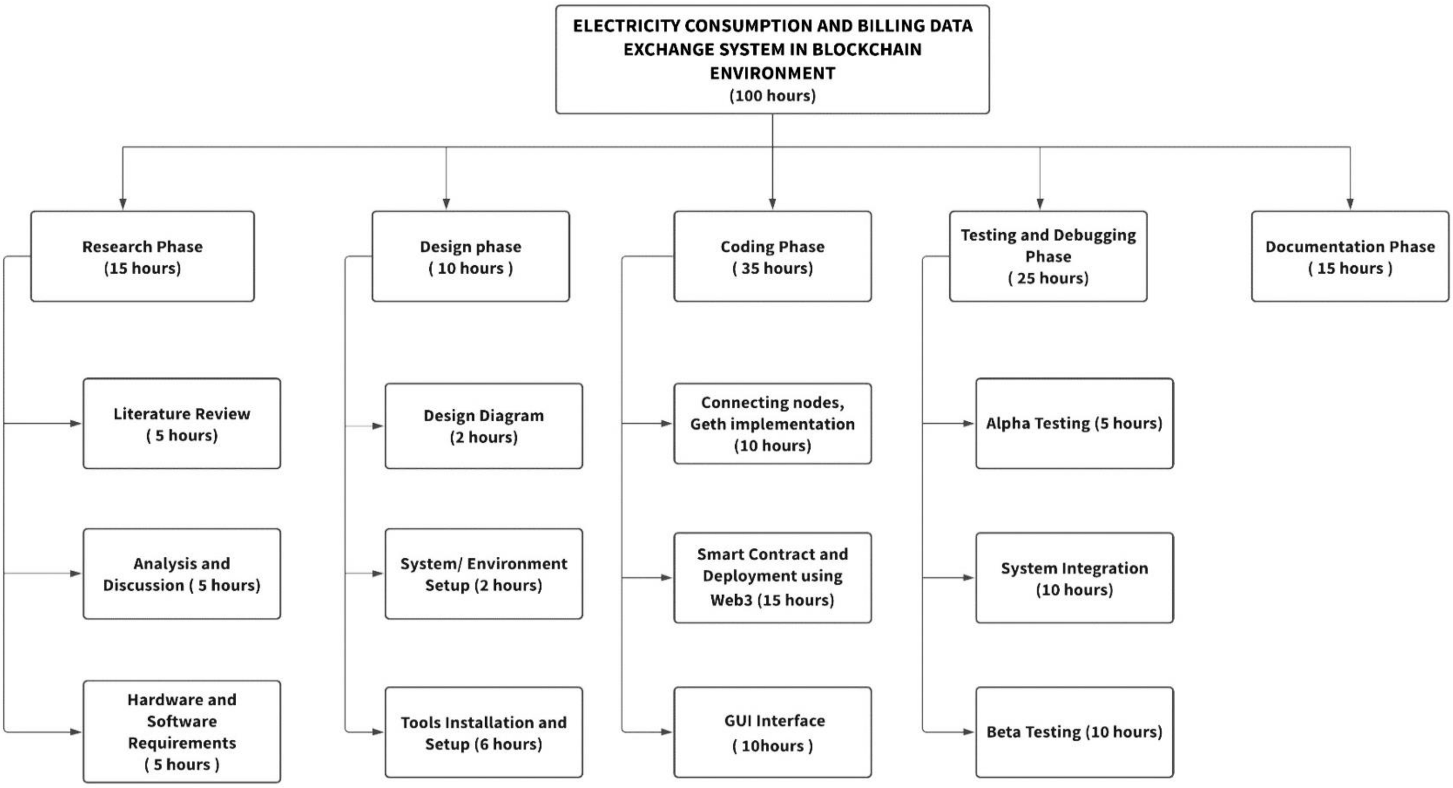

ELECTRICITY CONSUMPTION AND BILLING DATA EXCHANGE SYSTEM IN BLOCKCHAIN ENVIRONMENT (100 hours)
Research Phase (15 hours)
Design phase (10 hours)
Coding Phase (35 hours)
Testing and Debugging Phase (25 hours)
Documentation Phase (15 hours)
Literature Review (5 hours)
Analysis and Discussion (5 hours)
Hardware and Software Requirements (5 hours)
Design Diagram (2 hours)
System/ Environment Setup (2 hours)
Tools Installation and Setup (6 hours)
Connecting nodes, Geth implementation (10 hours)
Smart Contract and Deployment using Web3 (15 hours)
GUI Interface (10hours)
Alpha Testing (5 hours)
System Integration (10 hours)
Beta Testing (10 hours)

Fig1. Work Breakdown Structure for the Project

2.2.2.2 Gantt Chart: The Gantt Chart for our project is shown below with the start dates:

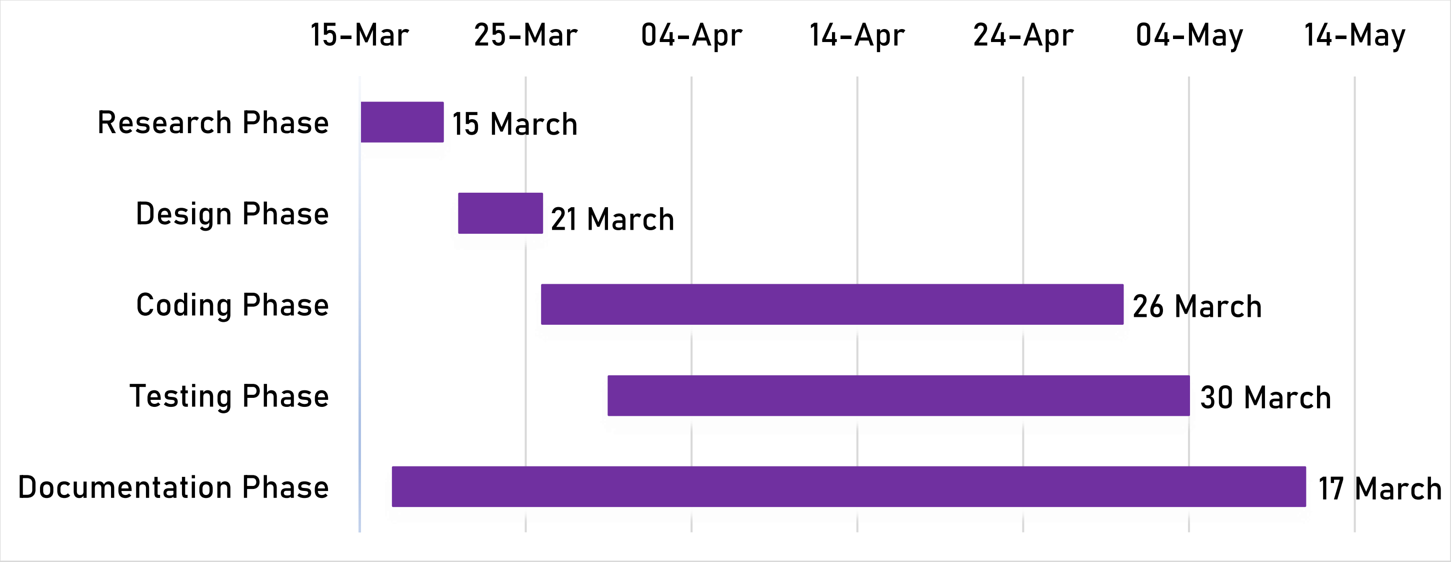


Fig.2 : Gantt Chart for the Project

2.2.3 Operational Feasibility: Proposed system is beneficial since it will prevent electricity board from generating incorrect bills and also prevent consumers from paying the same. This system provides transparency for both the ends and helps maintaining a clean billing record, which cannot be altered in anyway. Also, the final product developed can be used by any individual through a easy to use GUI.

2.2.4 Technical Feasibility: Technical feasibility centers on the existing computer system (hardware, software, etc.) and to what extent it can support the proposed addition. This project is based on blockchain. The technologies that are required are freely available in the internet for use. The limitations of the project and the ease of implementing are synchronized. The various tools and packages that will be used can be easily installed and configured, and execute smoothly on the machines. Moreover, the final product that will be generated should also work with ease on any Windows based platform. Therefore, our project is technically feasible.

CHAPTER 3

DESIGN DIAGRAMS

3.1 ACTIVITY DIAGRAM:

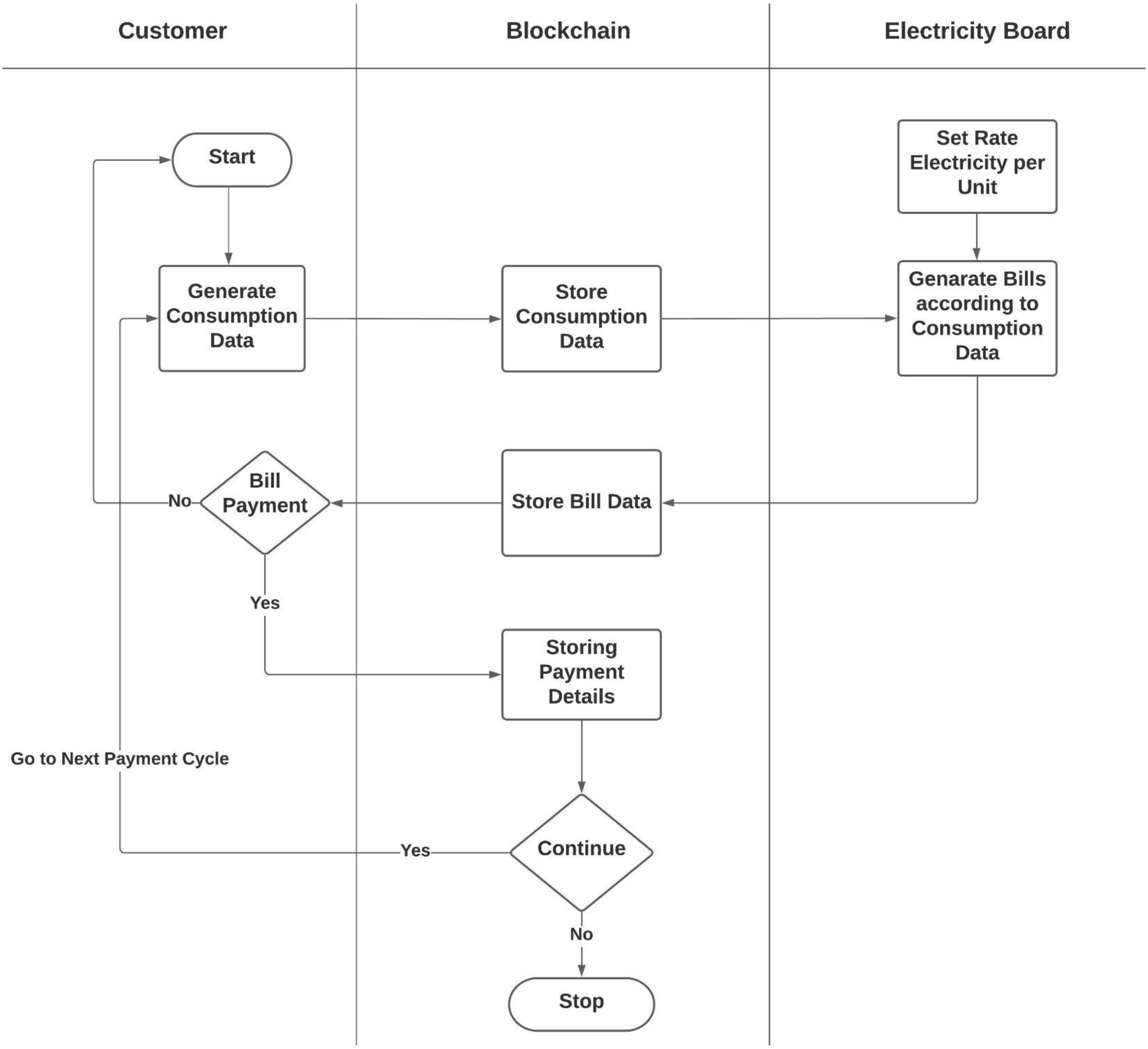


Fig 3: Activity Diagram for Electricity Consumption and Billing Data Exchange System in Blockchain Environment

3.2 Use Case Diagram:

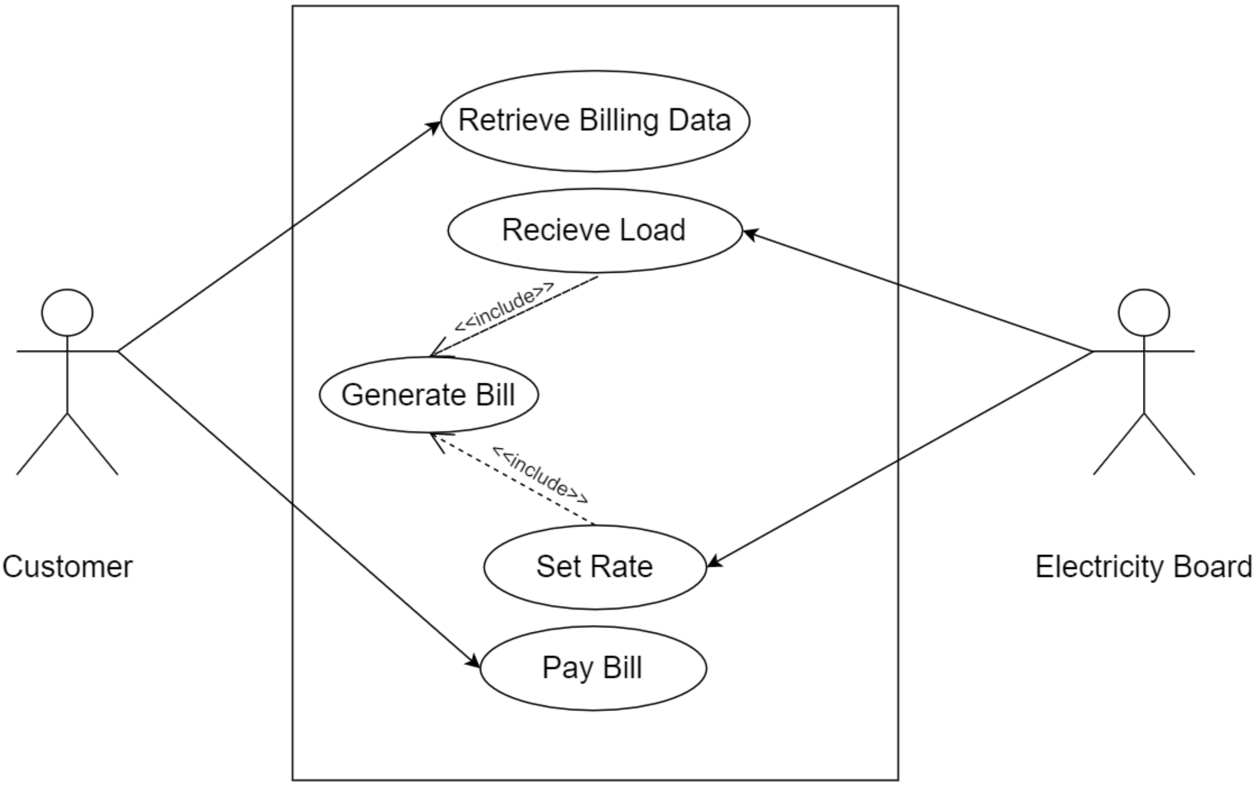


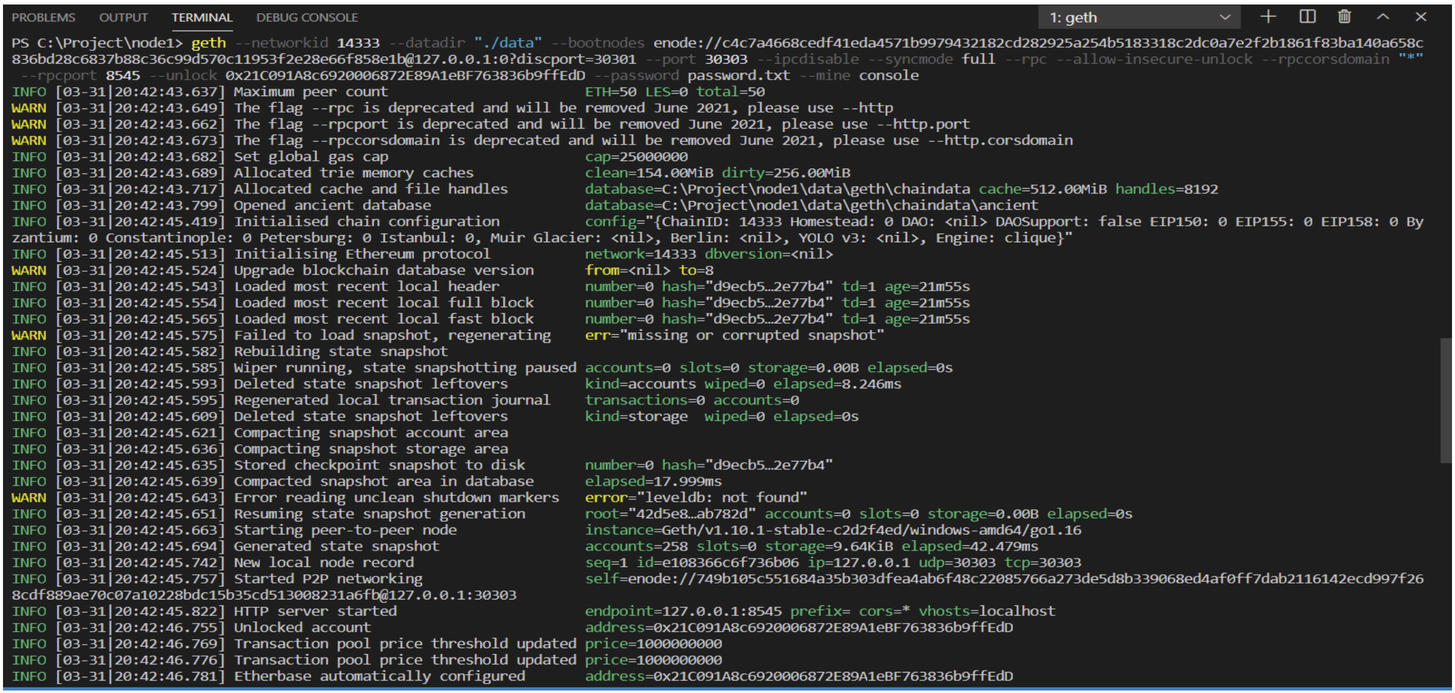
Fig 4: UML Use Case Diagram for Electricity Consumption and Billing Data Exchange System In Blockchain Environment

CHAPTER 4

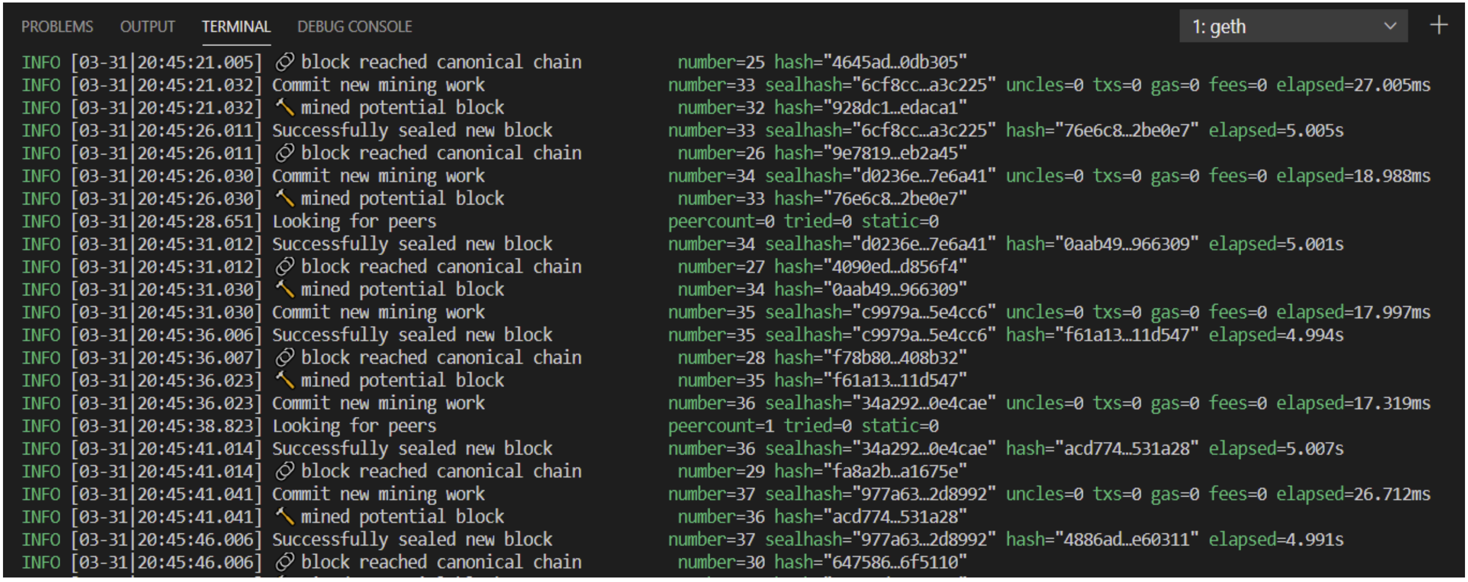
IMPLEMENTATION

As the project is based on Blockchain, so a private ethereum network was set up on our local machine to carry out various tasks that are required for the implementation and working of the system. Two nodes were created, one will act as the Electricity board and the other will act as the Consumer. Transactions will be carried among these two nodes within the private ethereum network.

To start the Ethereum network, we have to start the two nodes, and the bootnode together. The commands along with the running state of the nodes are shown below for node1:

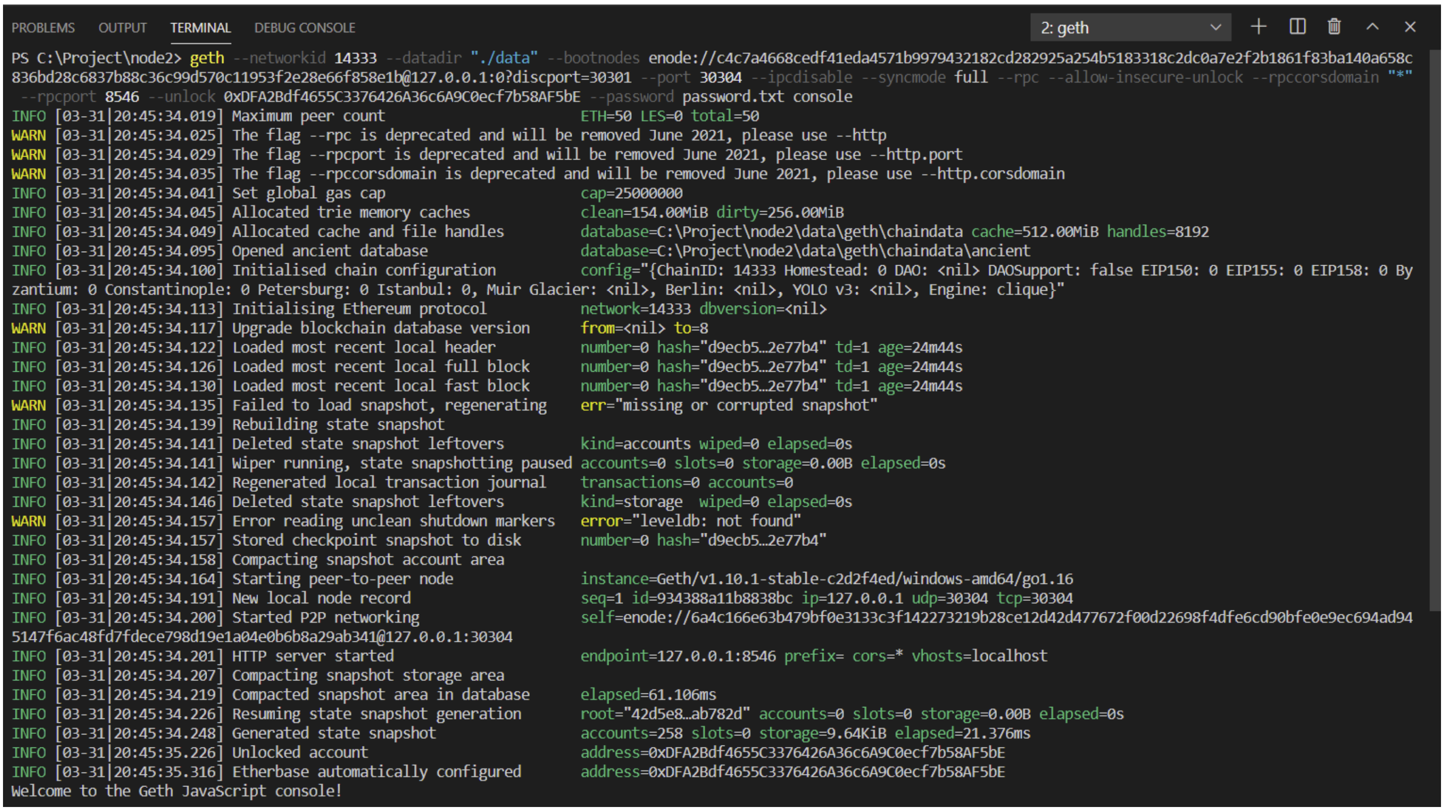


```
PS C:\Project\node1> geth --networkid 14333 --datadir "./data" --bootnodes enode://c4c7a4668cedf41eda4571b9979432182cd282925a254b5183318c2dc0a7e2f2b1861f83ba140a658c
836bd28c6837b88c36c99d570c11953f2e28e66f858e1b@127.0.0.1:0?discport=30301 --port 30303 --ipcdisable --syncmode full --rpc --allow-insecure-unlock --rpccorsdomain "*"
 --rpcport 8545 --unlock 0x21C091A8c6920006872E89A1eBF763836b9ffEdD --password password.txt --mine console
INFO [03-31|20:42:43.637] Maximum peer count                       ETH=50 LES=0 total=50
WARN [03-31|20:42:43.649] The flag --rpc is deprecated and will be removed June 2021, please use --http
WARN [03-31|20:42:43.662] The flag --rpcport is deprecated and will be removed June 2021, please use --http.port
WARN [03-31|20:42:43.673] The flag --rpccorsdomain is deprecated and will be removed June 2021, please use --http.corsdomain
INFO [03-31|20:42:43.682] Set global gas cap                       cap=25000000
INFO [03-31|20:42:43.689] Allocated trie memory caches             clean=154.00MiB dirty=256.00MiB
INFO [03-31|20:42:43.717] Allocated cache and file handles         database=C:\Project\node1\data\geth\chaindata cache=512.00MiB handles=8192
INFO [03-31|20:42:43.799] Opened ancient database                  database=C:\Project\node1\data\geth\chaindata\ancient
INFO [03-31|20:42:45.419] Initialised chain configuration          config="{ChainID: 14333 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: 0 EIP155: 0 EIP158: 0 By
zantium: 0 Constantinople: 0 Petersburg: 0 Istanbul: 0, Muir Glacier: <nil>, Berlin: <nil>, YOLO v3: <nil>, Engine: clique}"
INFO [03-31|20:42:45.513] Initialising Ethereum protocol           network=14333 dbversion=<nil>
WARN [03-31|20:42:45.524] Upgrade blockchain database version      from=<nil> to=8
INFO [03-31|20:42:45.543] Loaded most recent local header          number=0 hash="d9ecb5…2e77b4" td=1 age=21m55s
INFO [03-31|20:42:45.554] Loaded most recent local full block      number=0 hash="d9ecb5…2e77b4" td=1 age=21m55s
INFO [03-31|20:42:45.565] Loaded most recent local fast block      number=0 hash="d9ecb5…2e77b4" td=1 age=21m55s
WARN [03-31|20:42:45.575] Failed to load snapshot, regenerating    err="missing or corrupted snapshot"
INFO [03-31|20:42:45.582] Rebuilding state snapshot
INFO [03-31|20:42:45.585] Wiper running, state snapshotting paused accounts=0 slots=0 storage=0.00B elapsed=0s
INFO [03-31|20:42:45.593] Deleted state snapshot leftovers         kind=accounts wiped=0 elapsed=8.246ms
INFO [03-31|20:42:45.595] Regenerated local transaction journal    transactions=0 accounts=0
INFO [03-31|20:42:45.609] Deleted state snapshot leftovers         kind=storage  wiped=0 elapsed=0s
INFO [03-31|20:42:45.621] Compacting snapshot account area
INFO [03-31|20:42:45.636] Compacting snapshot storage area
INFO [03-31|20:42:45.635] Stored checkpoint snapshot to disk       number=0 hash="d9ecb5…2e77b4"
INFO [03-31|20:42:45.639] Compacted snapshot area in database      elapsed=17.999ms
WARN [03-31|20:42:45.643] Error reading unclean shutdown markers   error="leveldb: not found"
INFO [03-31|20:42:45.651] Resuming state snapshot generation       root="42d5e8…ab782d" accounts=0 slots=0 storage=0.00B elapsed=0s
INFO [03-31|20:42:45.663] Starting peer-to-peer node               instance=Geth/v1.10.1-stable-c2d2f4ed/windows-amd64/go1.16
INFO [03-31|20:42:45.694] Generated state snapshot                 accounts=258 slots=0 storage=9.64KiB elapsed=42.479ms
INFO [03-31|20:42:45.742] New local node record                    seq=1 id=e108366c6f736b06 ip=127.0.0.1 udp=30303 tcp=30303
INFO [03-31|20:42:45.757] Started P2P networking                   self=enode://749b105c551684a35b303dfea4ab6f48c22085766a273de5d8b339068ed4af0ff7dab2116142ecd997f26
8cdf889ae70c07a10228bdc15b35cd513008231a6fb@127.0.0.1:30303
INFO [03-31|20:42:45.822] HTTP server started                      endpoint=127.0.0.1:8545 prefix= cors=* vhosts=localhost
INFO [03-31|20:42:46.755] Unlocked account                         address=0x21C091A8c6920006872E89A1eBF763836b9ffEdD
INFO [03-31|20:42:46.769] Transaction pool price threshold updated price=1000000000
INFO [03-31|20:42:46.776] Transaction pool price threshold updated price=1000000000
INFO [03-31|20:42:46.781] Etherbase automatically configured       address=0x21C091A8c6920006872E89A1eBF763836b9ffEdD
```

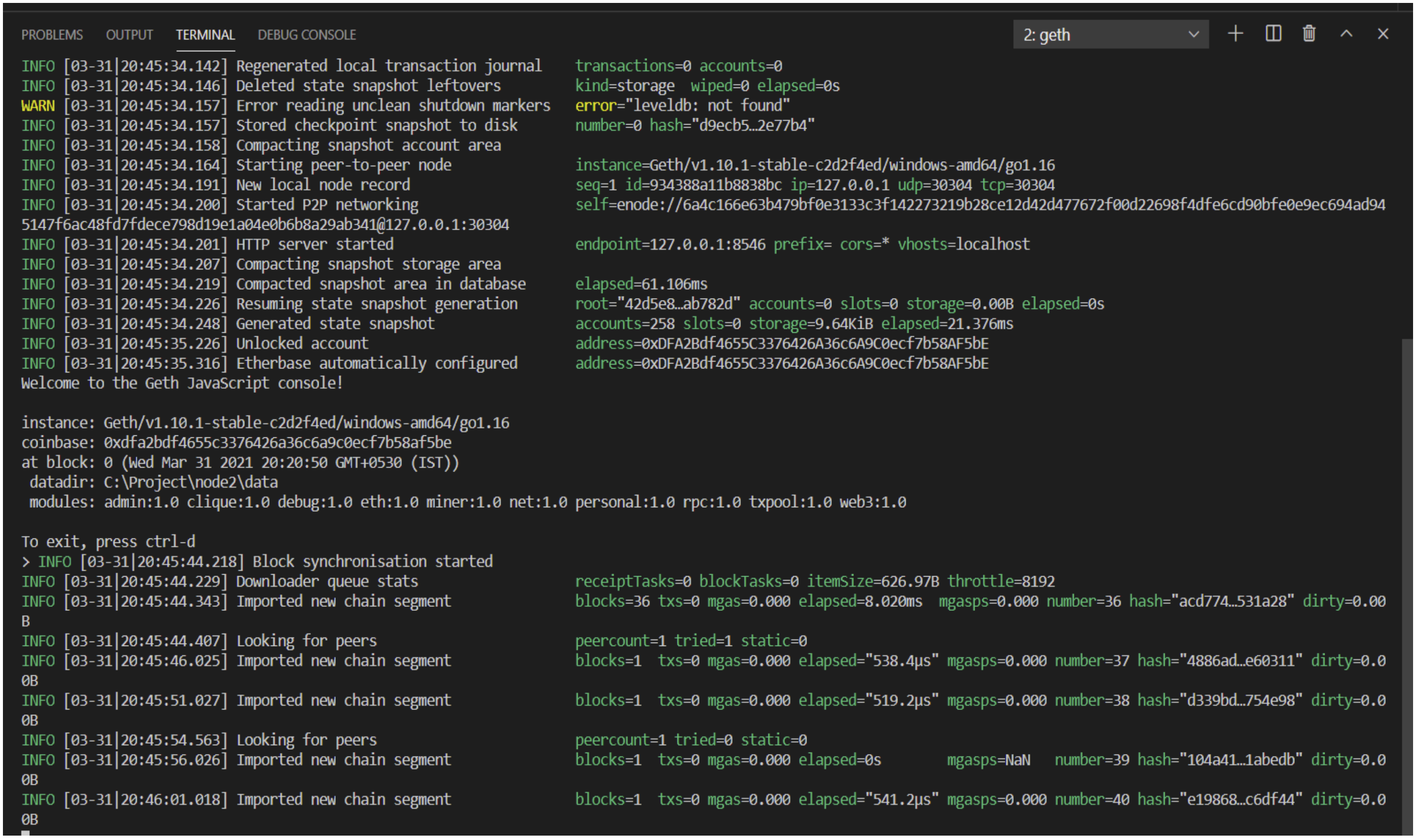


```
INFO [03-31|20:45:21.005] 🔗 block reached canonical chain          number=25 hash="4645ad…0db305"
INFO [03-31|20:45:21.032] Commit new mining work                   number=33 sealhash="6cf8cc…a3c225" uncles=0 txs=0 gas=0 fees=0 elapsed=27.005ms
INFO [03-31|20:45:21.032] 🔨 mined potential block                  number=32 hash="928dc1…edaca1"
INFO [03-31|20:45:26.011] Successfully sealed new block            number=33 sealhash="6cf8cc…a3c225" hash="76e6c8…2be0e7" elapsed=5.005s
INFO [03-31|20:45:26.011] 🔗 block reached canonical chain          number=26 hash="9e7819…eb2a45"
INFO [03-31|20:45:26.030] Commit new mining work                   number=34 sealhash="d0236e…7e6a41" uncles=0 txs=0 gas=0 fees=0 elapsed=18.988ms
INFO [03-31|20:45:26.030] 🔨 mined potential block                  number=33 hash="76e6c8…2be0e7"
INFO [03-31|20:45:28.651] Looking for peers                        peercount=0 tried=0 static=0
INFO [03-31|20:45:31.012] Successfully sealed new block            number=34 sealhash="d0236e…7e6a41" hash="0aab49…966309" elapsed=5.001s
INFO [03-31|20:45:31.012] 🔗 block reached canonical chain          number=27 hash="4090ed…d856f4"
INFO [03-31|20:45:31.030] 🔨 mined potential block                  number=34 hash="0aab49…966309"
INFO [03-31|20:45:31.030] Commit new mining work                   number=35 sealhash="c9979a…5e4cc6" uncles=0 txs=0 gas=0 fees=0 elapsed=17.997ms
INFO [03-31|20:45:36.006] Successfully sealed new block            number=35 sealhash="c9979a…5e4cc6" hash="f61a13…11d547" elapsed=4.994s
INFO [03-31|20:45:36.007] 🔗 block reached canonical chain          number=28 hash="f78b80…408b32"
INFO [03-31|20:45:36.023] 🔨 mined potential block                  number=35 hash="f61a13…11d547"
INFO [03-31|20:45:36.023] Commit new mining work                   number=36 sealhash="34a292…0e4cae" uncles=0 txs=0 gas=0 fees=0 elapsed=17.319ms
INFO [03-31|20:45:38.823] Looking for peers                        peercount=1 tried=0 static=0
INFO [03-31|20:45:41.014] Successfully sealed new block            number=36 sealhash="34a292…0e4cae" hash="acd774…531a28" elapsed=5.007s
INFO [03-31|20:45:41.014] 🔗 block reached canonical chain          number=29 hash="fa8a2b…a1675e"
INFO [03-31|20:45:41.041] Commit new mining work                   number=37 sealhash="977a63…2d8992" uncles=0 txs=0 gas=0 fees=0 elapsed=26.712ms
INFO [03-31|20:45:41.041] 🔨 mined potential block                  number=36 hash="acd774…531a28"
INFO [03-31|20:45:46.006] Successfully sealed new block            number=37 sealhash="977a63…2d8992" hash="4886ad…e60311" elapsed=4.991s
INFO [03-31|20:45:46.006] 🔗 block reached canonical chain          number=30 hash="647586…6f5110"
```

Similarly, for node2:



```
PS C:\Project\node2> geth --networkid 14333 --datadir "./data" --bootnodes enode://c4c7a4668cedf41eda4571b9979432182cd282925a254b5183318c2dc0a7e2f2b1861f83ba140a658c
836bd28c6837b88c36c99d570c11953f2e28e66f858e1b@127.0.0.1:0?discport=30301 --port 30304 --ipcdisable --syncmode full --rpc --allow-insecure-unlock --rpccorsdomain "*"
 --rpcport 8546 --unlock 0xDFA2Bdf4655C3376426A36c6A9C0ecf7b58AF5bE --password password.txt console
INFO [03-31|20:45:34.019] Maximum peer count                       ETH=50 LES=0 total=50
WARN [03-31|20:45:34.025] The flag --rpc is deprecated and will be removed June 2021, please use --http
WARN [03-31|20:45:34.029] The flag --rpcport is deprecated and will be removed June 2021, please use --http.port
WARN [03-31|20:45:34.035] The flag --rpccorsdomain is deprecated and will be removed June 2021, please use --http.corsdomain
INFO [03-31|20:45:34.041] Set global gas cap                       cap=25000000
INFO [03-31|20:45:34.045] Allocated trie memory caches             clean=154.00MiB dirty=256.00MiB
INFO [03-31|20:45:34.049] Allocated cache and file handles         database=C:\Project\node2\data\geth\chaindata cache=512.00MiB handles=8192
INFO [03-31|20:45:34.095] Opened ancient database                  database=C:\Project\node2\data\geth\chaindata\ancient
INFO [03-31|20:45:34.100] Initialised chain configuration          config="{ChainID: 14333 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: 0 EIP155: 0 EIP158: 0 By
zantium: 0 Constantinople: 0 Petersburg: 0 Istanbul: 0, Muir Glacier: <nil>, Berlin: <nil>, YOLO v3: <nil>, Engine: clique}"
INFO [03-31|20:45:34.113] Initialising Ethereum protocol           network=14333 dbversion=<nil>
WARN [03-31|20:45:34.117] Upgrade blockchain database version      from=<nil> to=8
INFO [03-31|20:45:34.122] Loaded most recent local header          number=0 hash="d9ecb5…2e77b4" td=1 age=24m44s
INFO [03-31|20:45:34.126] Loaded most recent local full block      number=0 hash="d9ecb5…2e77b4" td=1 age=24m44s
INFO [03-31|20:45:34.130] Loaded most recent local fast block      number=0 hash="d9ecb5…2e77b4" td=1 age=24m44s
WARN [03-31|20:45:34.135] Failed to load snapshot, regenerating    err="missing or corrupted snapshot"
INFO [03-31|20:45:34.139] Rebuilding state snapshot
INFO [03-31|20:45:34.141] Deleted state snapshot leftovers         kind=accounts wiped=0 elapsed=0s
INFO [03-31|20:45:34.141] Wiper running, state snapshotting paused accounts=0 slots=0 storage=0.00B elapsed=0s
INFO [03-31|20:45:34.142] Regenerated local transaction journal    transactions=0 accounts=0
INFO [03-31|20:45:34.146] Deleted state snapshot leftovers         kind=storage  wiped=0 elapsed=0s
WARN [03-31|20:45:34.157] Error reading unclean shutdown markers   error="leveldb: not found"
INFO [03-31|20:45:34.157] Stored checkpoint snapshot to disk       number=0 hash="d9ecb5…2e77b4"
INFO [03-31|20:45:34.158] Compacting snapshot account area
INFO [03-31|20:45:34.164] Starting peer-to-peer node               instance=Geth/v1.10.1-stable-c2d2f4ed/windows-amd64/go1.16
INFO [03-31|20:45:34.191] New local node record                    seq=1 id=934388a11b8838bc ip=127.0.0.1 udp=30304 tcp=30304
INFO [03-31|20:45:34.200] Started P2P networking                   self=enode://6a4c166e63b479bf0e3133c3f142273219b28ce12d42d477672f00d22698f4dfe6cd90bfe0e9ec694ad94
5147f6ac48fd7fdece798d19e1a04e0b6b8a29ab341@127.0.0.1:30304
INFO [03-31|20:45:34.201] HTTP server started                      endpoint=127.0.0.1:8546 prefix= cors=* vhosts=localhost
INFO [03-31|20:45:34.207] Compacting snapshot storage area
INFO [03-31|20:45:34.219] Compacted snapshot area in database      elapsed=61.106ms
INFO [03-31|20:45:34.226] Resuming state snapshot generation       root="42d5e8…ab782d" accounts=0 slots=0 storage=0.00B elapsed=0s
INFO [03-31|20:45:34.248] Generated state snapshot                 accounts=258 slots=0 storage=9.64KiB elapsed=21.376ms
INFO [03-31|20:45:35.226] Unlocked account                         address=0xDFA2Bdf4655C3376426A36c6A9C0ecf7b58AF5bE
INFO [03-31|20:45:35.316] Etherbase automatically configured       address=0xDFA2Bdf4655C3376426A36c6A9C0ecf7b58AF5bE
Welcome to the Geth JavaScript console!
```

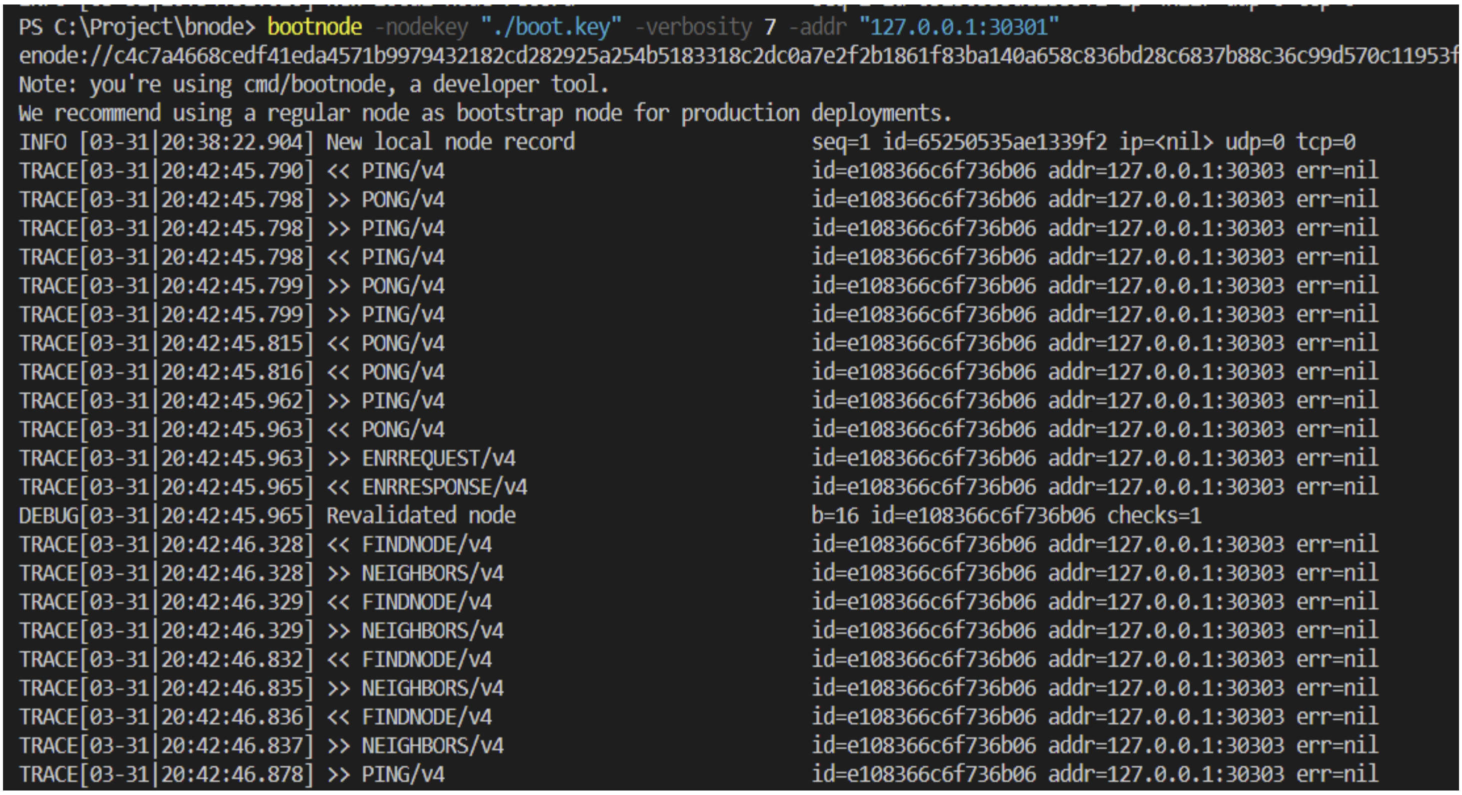


```
INFO [03-31|20:45:34.142] Regenerated local transaction journal    transactions=0 accounts=0
INFO [03-31|20:45:34.146] Deleted state snapshot leftovers         kind=storage  wiped=0 elapsed=0s
WARN [03-31|20:45:34.157] Error reading unclean shutdown markers   error="leveldb: not found"
INFO [03-31|20:45:34.157] Stored checkpoint snapshot to disk       number=0 hash="d9ecb5…2e77b4"
INFO [03-31|20:45:34.158] Compacting snapshot account area
INFO [03-31|20:45:34.164] Starting peer-to-peer node               instance=Geth/v1.10.1-stable-c2d2f4ed/windows-amd64/go1.16
INFO [03-31|20:45:34.191] New local node record                    seq=1 id=934388a11b8838bc ip=127.0.0.1 udp=30304 tcp=30304
INFO [03-31|20:45:34.200] Started P2P networking                   self=enode://6a4c166e63b479bf0e3133c3f142273219b28ce12d42d477672f00d22698f4dfe6cd90bfe0e9ec694ad94
5147f6ac48fd7fdece798d19e1a04e0b6b8a29ab341@127.0.0.1:30304
INFO [03-31|20:45:34.201] HTTP server started                      endpoint=127.0.0.1:8546 prefix= cors=* vhosts=localhost
INFO [03-31|20:45:34.207] Compacting snapshot storage area
INFO [03-31|20:45:34.219] Compacted snapshot area in database      elapsed=61.106ms
INFO [03-31|20:45:34.226] Resuming state snapshot generation       root="42d5e8…ab782d" accounts=0 slots=0 storage=0.00B elapsed=0s
INFO [03-31|20:45:34.248] Generated state snapshot                 accounts=258 slots=0 storage=9.64KiB elapsed=21.376ms
INFO [03-31|20:45:35.226] Unlocked account                         address=0xDFA2Bdf4655C3376426A36c6A9C0ecf7b58AF5bE
INFO [03-31|20:45:35.316] Etherbase automatically configured       address=0xDFA2Bdf4655C3376426A36c6A9C0ecf7b58AF5bE
Welcome to the Geth JavaScript console!

instance: Geth/v1.10.1-stable-c2d2f4ed/windows-amd64/go1.16
coinbase: 0xdfa2bdf4655c3376426a36c6a9c0ecf7b58af5be
at block: 0 (Wed Mar 31 2021 20:20:50 GMT+0530 (IST))
 datadir: C:\Project\node2\data
 modules: admin:1.0 clique:1.0 debug:1.0 eth:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

To exit, press ctrl-d
> INFO [03-31|20:45:44.218] Block synchronisation started
INFO [03-31|20:45:44.229] Downloader queue stats                   receiptTasks=0 blockTasks=0 itemSize=626.97B throttle=8192
INFO [03-31|20:45:44.343] Imported new chain segment               blocks=36 txs=0 mgas=0.000 elapsed=8.020ms  mgasps=0.000 number=36 hash="acd774…531a28" dirty=0.00
B
INFO [03-31|20:45:44.407] Looking for peers                        peercount=1 tried=1 static=0
INFO [03-31|20:45:46.025] Imported new chain segment               blocks=1  txs=0 mgas=0.000 elapsed="538.4µs" mgasps=0.000 number=37 hash="4886ad…e60311" dirty=0.0
0B
INFO [03-31|20:45:51.027] Imported new chain segment               blocks=1  txs=0 mgas=0.000 elapsed="519.2µs" mgasps=0.000 number=38 hash="d339bd…754e98" dirty=0.0
0B
INFO [03-31|20:45:54.563] Looking for peers                        peercount=1 tried=0 static=0
INFO [03-31|20:45:56.026] Imported new chain segment               blocks=1  txs=0 mgas=0.000 elapsed=0s       mgasps=NaN   number=39 hash="104a41…1abedb" dirty=0.0
0B
INFO [03-31|20:46:01.018] Imported new chain segment               blocks=1  txs=0 mgas=0.000 elapsed="541.2µs" mgasps=0.000 number=40 hash="e19868…c6df44" dirty=0.0
0B
```

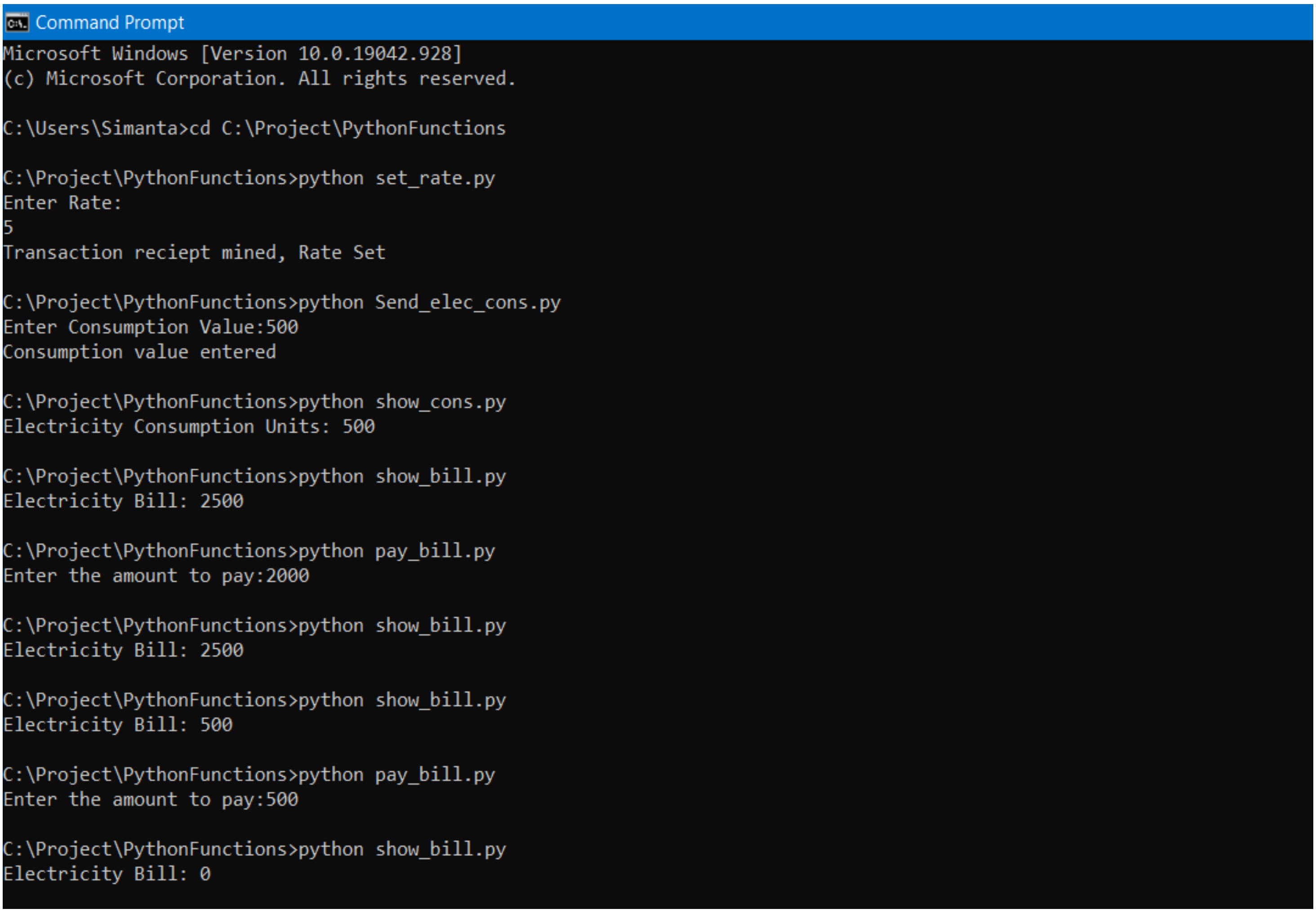
For the bootnode:



```
PS C:\Project\bnode> bootnode -nodekey "./boot.key" -verbosity 7 -addr "127.0.0.1:30301"
enode://c4c7a4668cedf41eda4571b9979432182cd282925a254b5183318c2dc0a7e2f2b1861f83ba140a658c836bd28c6837b88c36c99d570c11953f
Note: you're using cmd/bootnode, a developer tool.
We recommend using a regular node as bootstrap node for production deployments.
INFO [03-31|20:38:22.904] New local node record                    seq=1 id=65250535ae1339f2 ip=<nil> udp=0 tcp=0
TRACE[03-31|20:42:45.790] << PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.798] >> PONG/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.798] >> PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.798] << PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.799] >> PONG/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.799] >> PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.815] << PONG/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.816] << PONG/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.962] >> PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.963] << PONG/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.963] >> ENRREQUEST/v4                         id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:45.965] << ENRRESPONSE/v4                        id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
DEBUG[03-31|20:42:45.965] Revalidated node                         b=16 id=e108366c6f736b06 checks=1
TRACE[03-31|20:42:46.328] << FINDNODE/v4                           id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.328] >> NEIGHBORS/v4                          id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.329] << FINDNODE/v4                           id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.329] >> NEIGHBORS/v4                          id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.832] << FINDNODE/v4                           id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.835] >> NEIGHBORS/v4                          id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.836] << FINDNODE/v4                           id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.837] >> NEIGHBORS/v4                          id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
TRACE[03-31|20:42:46.878] >> PING/v4                               id=e108366c6f736b06 addr=127.0.0.1:30303 err=nil
```

A smart contract was written after the setup of Ethereum network. It was compiled and deployed successfully using remix Ethereum for the first time. Later, python programs are written to call the functions inside the smart contract to carry out the input and output of data in the blockchain.

The following screenshot shows the usage of the functions of smart contract that are used to send electricity consumption data, view the consumption data, set rate per unit, pay bill etc.



```
Microsoft Windows [Version 10.0.19042.928]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Simanta>cd C:\Project\PythonFunctions

C:\Project\PythonFunctions>python set_rate.py
Enter Rate:
5
Transaction reciept mined, Rate Set

C:\Project\PythonFunctions>python Send_elec_cons.py
Enter Consumption Value:500
Consumption value entered

C:\Project\PythonFunctions>python show_cons.py
Electricity Consumption Units: 500

C:\Project\PythonFunctions>python show_bill.py
Electricity Bill: 2500

C:\Project\PythonFunctions>python pay_bill.py
Enter the amount to pay:2000

C:\Project\PythonFunctions>python show_bill.py
Electricity Bill: 2500

C:\Project\PythonFunctions>python show_bill.py
Electricity Bill: 500

C:\Project\PythonFunctions>python pay_bill.py
Enter the amount to pay:500

C:\Project\PythonFunctions>python show_bill.py
Electricity Bill: 0
```

CONCLUSION

This will be a python based, GUI application that will interact with the Ethereum network to process the billing. This project relies on the blockchain technology to operate depending on the user inputs and the functions that will be inside the smart contract. Web3 library functions are used to interact and deploy the smart contracts.

REFERENCES

1. Gür AÖ, Öksüzer Ş, Karaarslan E. Blockchain based metering and billing system proposal with privacy protection for the electric network. In2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG) 2019 Apr 25 (pp. 204-208). Ieee.
2. Kumar T, Ramani V, Ahmad I, Braeken A, Harjula E, Ylianttila M. Blockchain utilization in healthcare: Key requirements and challenges. In2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom) 2018 Sep 17 (pp. 1-7). IEEE.
3. Maksymyuk T, Gazda J, Han L, Jo M. Blockchain-based intelligent network management for 5G and beyond. In2019 3rd International Conference on Advanced Information and Communications Technologies (AICT) 2019 Jul 2 (pp. 36-39). IEEE.
4. Kim NH, Kang SM, Hong CS. Mobile charger billing system using lightweight Blockchain. In2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS) 2017 Sep 27 (pp. 374-377). IEEE.
5. Geth Documentation : https://geth.ethereum.org/docs/
   Last accessed: 20-March, 2021
6. Web3.py Documentation : https://web3py.readthedocs.io/en/stable/
   Last accessed: 15-March, 2021
7. Solidity Documentation : https://docs.soliditylang.org/en/v0.8.2/
   Last accessed: 18-March, 2021