A Project report on

"E-BIDDING"

Submitted by:

Shweta H. Chudasama - 1698002

Vidhi D. Jain - 1698011

Ankita V. Tripathi - 1698037

Guided by:

Mrs. Radhika Patwardhan

Term:

December '18 to April '19

Department of Information Technology

SVKM's Shri Bhagubhai Mafatlal Polytechnic,

Irla, Vile Parle (West), Mumbai – 400 056.

ACKNOWLEDGEMENT

Acknowledgements

We take this opportunity to express our profound gratitude and deep regards to our guide Mrs. Radhika Patwardhan for her exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessings, help and guidance by her 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 Mr Abhijit Dongaonkar (Head of IT Department, SBMP), Mrs Neeta Kadukar (Vice Principal, SBMP) for your coordinal support, valuable information and guidance which helped us in completing this task through various stages.

Also we would like to thank Mr Abhishek Chande for guiding us and explaining us the working of the organization and process of releasing tender and associated functionality such as (PO) purchase order, (RFQ) Request for Query, (RFP) Request for proposal, Screening and evaluation etc. which helped us in understanding the working of organizations, how tender is being sent and received.

We are obliged to staff members and technicians of IT department for the valuable information provided by them in their respective fields. We are grateful for their cooperation during the period of our project. Lastly, we thank almighty, our parents and our classmates for their constant encouragement without which this project would not have been possible

ABSTRACT

Abstract

In the real world we have seen many auctions happening manually for selling particular property wherein both the seller and buyers need to be present at the same time which is not possible and also time-consuming now-a-days in this busy world.

E-Bidding application solves the problem that is mentioned above by saving time of both the buyers and the sellers. This application is developed for auction of any of the car's spare parts. This application provides login for both buyer and seller. They can access the application at any time and from anywhere. Seller and Bidder will have their own blocks. Seller can add the spare parts of car that are required for an organization and specify the time-limit for it. The buyers can start bidding for the products by specifying desired bidding price. This application allows tender to be released, Request for Query (RFQ) by sellers and bidders will provide their Request for Proposal (RFP). The sellers will then confirm the bid by selecting the lowest tender amount offered by bidders in a specified amount of time. The seller can also view the results. Both of them can edit their profiles. The bidders will be informed which proposal has been selected by seller. What is the final bid amount for the product? This project is developed to solve difficulties faced by the Car Companies for obtaining spare parts from manufacturing companies at low price and best quality which was done manually, allowing them to release tender and make bidding process for providing spare parts easier by removing third-party involvement and increasing security with the help of Blockchain technology which is feasible for any kind of browsers, processor, Operating Systems (only Windows) and memory used.

Index

INTRODUCTION

1. Introduction

Web applications have been around since before the World Wide Web gained mainstream popularity. A web application (or "web app" for short) is any computer program that performs a specific function by using a web browser as its client. The application can be as simple as a message board or a contact form on a website or as complex as a word processor or a multi-player mobile gaming app that you download to your phone. A web application relieves the developer of the responsibility of building a client for a specific type of computer or a specific operating system, so anyone can use the application along as they have internet access. Web applications can provide the same functionality and gain the benefit of working across multiple platforms. As compared to the desktop applications, these are cost-effective. At the same time, they do not waste any space on the computer, as the software runs from a server. One does not have to pay for maintenance fees either. Web applications can play a crucial role in the branding process. With their help, it is easier to maintain a proper communication channel between potential customers and the business organization. Web Applications can even work on multiple platforms and they can be directly accessed.

The goal of blockchain technology is to make for a *technology* that enables transparency, security, and permanence in validating and recording transactions. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a merkle tree root hash).A blockchain is a peer-to-peer network of computers, called nodes that share all the data and the code in the network. So, if a device is connected to the blockchain, then it is a node in the network, and it can talk to all the other computer nodes in the network. Ethereum is a distributed public blockchain network. Although there are some significant technical differences between the two, the most important distinction to note is that Bitcoin and Ethereum differ substantially in purpose and capability. The Ethereum blockchain, instead of mining for bitcoin, miners work to earn Ether, a type of crypto token that fuels the network. In the Ethereum blockchain, instead of mining for bitcoin, miners work to earn Ether, a type of crypto token that fuels the network. Beyond a tradeable cryptocurrency, Ether is also used by application developers to pay for transaction fees and services on the Ethereum network. Accordingly Ethereum, it can be used to codify, decentralize, secure and trade just about anything.

By considering the above advantages of web applications, theory of Ethereum and goals of Blockchain technology we have developed a web application which provides decentralized infrastructure for auctions of electronic products helping buyers and sellers around the world to communicate and work effectively.

PROBLEM DEFINITION

2. Problem Definition

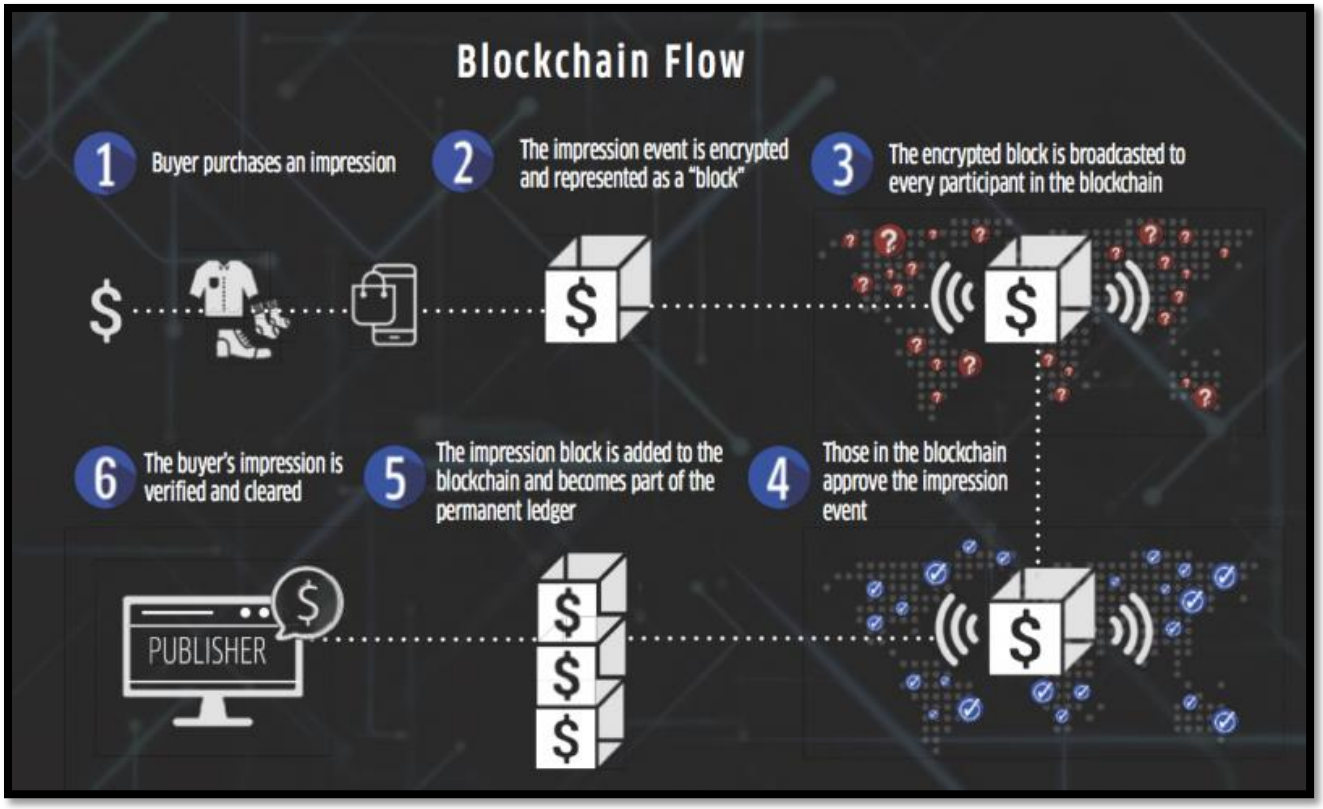
The era of auction through websites opens the windows to the technology-oriented auctions. It's the time to change from conventional auctions to automated auctions, which has become important now-a-days.
Dramatic breakthroughs in processing power along with the number of extra features included in web applications have opened the doors to a wide range of commercial, governmental as well as educational possibilities.

The earlier auctions websites were used for selling general products. They were not much secure, required third party authentication and could be violated by anyone. Governments, Organizations and Companies were required to provide tender after certain limit for any kind of work. They used to specify their requirements for spare parts to the manufacturing companies; tenders were brought; selected manually by the company which was time consuming wherein both seller and buyer need to be present at the same time which is not possible. It required third-party authentication which was perilous as information could be hacked by the companies for their ravenous needs which resulted in losses to the company because tender is legal document so security was easily breached creating crux throughout the organizations, companies ,governments etc. Also they faced the problem of visiting the company again and again until tender was passed, all this would take even week for sending RFQ, RFP, PO along with screening and evaluation.

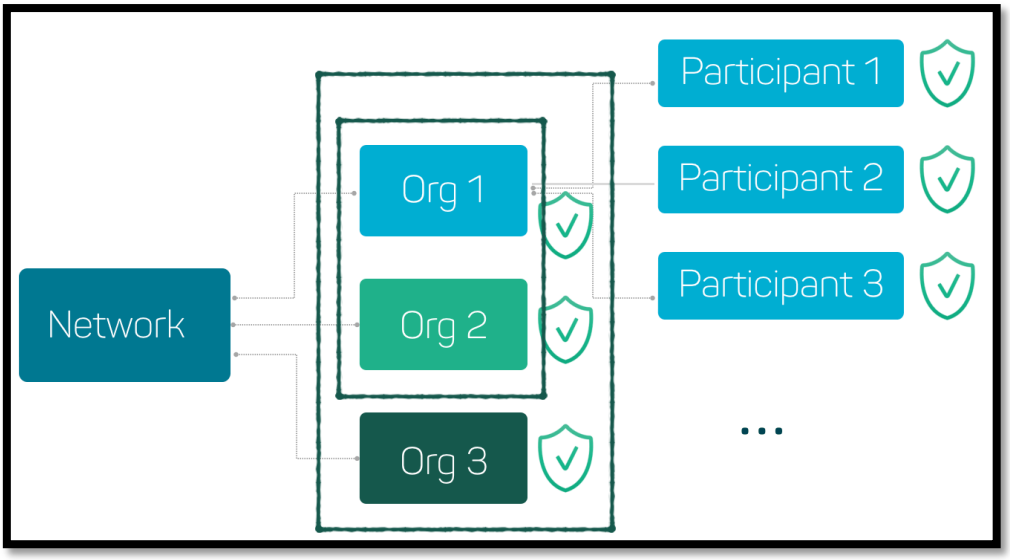
PROPOSED SYSTEM

3. Proposed System

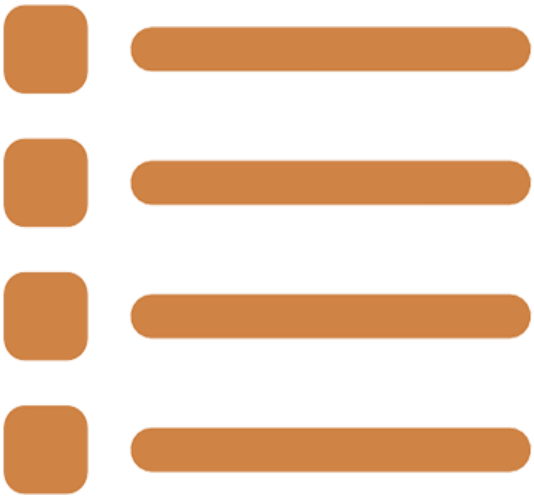
We have proposed in this project a web application as it relieves the developer of the responsibility of building a client for a specific type of computer or a specific operating system, so anyone can use the application along as they have internet access using Blockchain technology which is designed for organizations selling spare car products. Blockchain is one of the most popular technologies as it provides trusted third-party to facilitate our transactions and added security along with distributed infrastructure.



In this application we will provide login for both buyer and seller. Sellers and Bidders can access this application anytime and anywhere. Seller and Bidder will have their own blocks. This application solves the problem discussed above by allowing seller to add the RFQ (Request for Query) for spare parts required by the Car Company and specifies the requirements along-with time-limit for it which is done through this application remotely. The buyers can bid for the spare parts by releasing their tenders i.e. Request for Proposal (RFP) and the desired biding price. The seller can confirm the bid by selecting the lowest bid amount and best quality offered by bidders in specified time period. The seller can also view the results. Both of them will be allowed to edit their profiles. This application will provide sellers and buyers to view the spare parts, price, and results. Security is provided by hashing passwords entered by buyer and seller so that no hacker can violate the privacy of users.



The flexibility of adding products dynamically on runtime along with timer is also provided which is visible through remote access by using lite-server. This application also provides functionality of sending private key through email and also to send notification to users when they forget the password. Thus this application will help organization in need of such applications for conducting auctions.



FEATURES

4. Features

- Auction process is simplified.

  This feature aims to provide facility for conducting auction to people anywhere in the world by using Blockchain technology which is more secure eliminating the need for third-party authentication.

- Login and registration.

  This feature aims to provide login and registration to sellers and bidders verified accordingly by the system.

- Bidders can bid for products.

  This feature aims to provide bidders with the facility to bid for sear parts being released as tender by sellers of a particular Car Company within time-limit.

- User-friendly.

  This feature aims to allow this application to be accessed by any people anytime, anywhere without being available at the same place due to busy life.

- Security is maintained by hashing passwords.

  This feature aims to provide additional security by hashing passwords entered by sellers and bidders as a part of identity so that no hacker can hack the password or violate the privacy of users.

- Proof of work is used.

  This feature aims to maintain integrity of data stored in blocks by hashing data and keeping hash of previous block to provide chain of blocks so that no third person can enter the system and if so proof of work is used to verify users in network.

- Sellers can add RFQ.

  This feature allows sellers to add RFQ on the basis of needs of Car Company and also specify time-limit within which the bidders must bid for tenders.

  - Bidders can add RFP

This feature aims to allow bidders to add RFP based on requirements of seller for spare parts as it is legal document so credentials is securely passed without displaying it to others.

- Profiles can be edited.

  This feature allows sellers and bidders to edit their profiles accordingly.

- Tender is released.

  This feature aims to allow bidders to add and release tender for products required by sellers of Car Company and bidder with lowest price and best quality will be given contract after performing the process of screening and evaluation.

- User cannot go back

  This feature prevents users from going back after performing any operations in the system so no modifications are allowed.

- Results are displayed.

  In this feature the bidder who wins will be shown along with others who participated in auction by displaying price which seller can view.

- Passwords are hashed.

  By hashing passwords entered by users additional security can be provided and hackers ultimatum is also reduced by verifying hash of passwords entered during the time of registration

SOFTWARE REQUIREMENT

5. Software Requirements

Back-end-

- Node-js:

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js's package ecosystem, NPM, is the largest ecosystem of open source libraries in the world. For our project Node.js will be the main framework on which the server will be built all other backend technologies act as add-ons or packages that are added as modules to the node.js framework.

- Solidity:

Solidity is known as a contract-based, high-level programming language. This platform has similar syntax to the scripting language of JavaScript. Solidity as a programming language is made to enhance the Ethereum Virtual Machine. Solidity is statically typed scripting language which does the process of verifying and enforcing the constraints at compile-time as opposed to run-time.

- Metamask:

MetaMask is a browser plugin that allows users to make Ethereum transactions through regular websites. It facilitates the adoption of Ethereum because it bridges the gap between the user interfaces for Ethereum. Without this, Ethereum cannot be considered mainstream because the regular web has a strong network effect which inhibits the average user from switching. MetaMask manages your Ethereum wallet, which contains your Ethers (or money), and allows you to send and receive Ethers through a dApps of interest.

- Ganache:

Ganache, previously Testrpc, is a virtual blockchain which sets up 10 default Ethereum addresses, complete with private keys and all, and pre-loads them with 100 simulated Ether each. There is no "mining" per-se with Ganache - instead, it immediately confirms any transaction coming its way. This makes iterative development possible - you can write unit tests for your code which execute on this simulated blockchain, deploy smart contracts, play around, call

functions, and then tear it all down for further simulation or new tests, returning all addresses to their initial state of 100 Ether.

- Truffle :

Truffle is a development environment, testing framework and asset pipeline for Ethereum, aiming to make life as an Ethereum developer easier. It is one of the most widely used IDEs in the Ethereum community. Developers can use it to build and deploy DApps for testing purposes with many features that make it more attractive to users with a Web 3.0 dev background.

- Lite-server:

Lightweight development only node server that serves a web app, opens it in the browser, refreshes when html or Javascript change, injects CSS changes using sockets, and has a fall back page when a route is not found. Browser Sync does most of what we want in a super-fast lightweight development server. It serves the static content, detects changes, refreshes the browsers, and offers many customizations. Command is:-

npm install lite-server

Front-end-

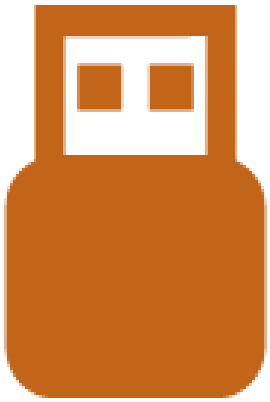
- HTML, CSS :

Html is a Hypertext Markup Language for creating a web page. **HTML** is used to mark and describe each of these kinds of content so the web browser can display them correctly. HTML code ensures the proper formatting of text and images so that your Internet browser may display them as they are intended to look.

Cascading Style Sheets are used for styling the web page and m more attractive and simplifies the process of making them presentable.

- JavaScript:

JavaScript is the programming language for the web and used to update HTML and CSS. Javascript can calculate, validate and manipulate data. It is both server-side and client-side scripting language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is designed for creating network-centric applications. Also it is complementary to and integrated with Java.



HARDWARE REQUIREMENTS

6. Hardware Requirements

- Back-end:-

RAM: _GB

Secondary Storage: _GB

Processor: 2 GHz +

- **Front-end:-**

a) Web Browser:

Google Chrome 70+/latest (recommended) or

Mozilla Firefox 50+ or Opera 42+

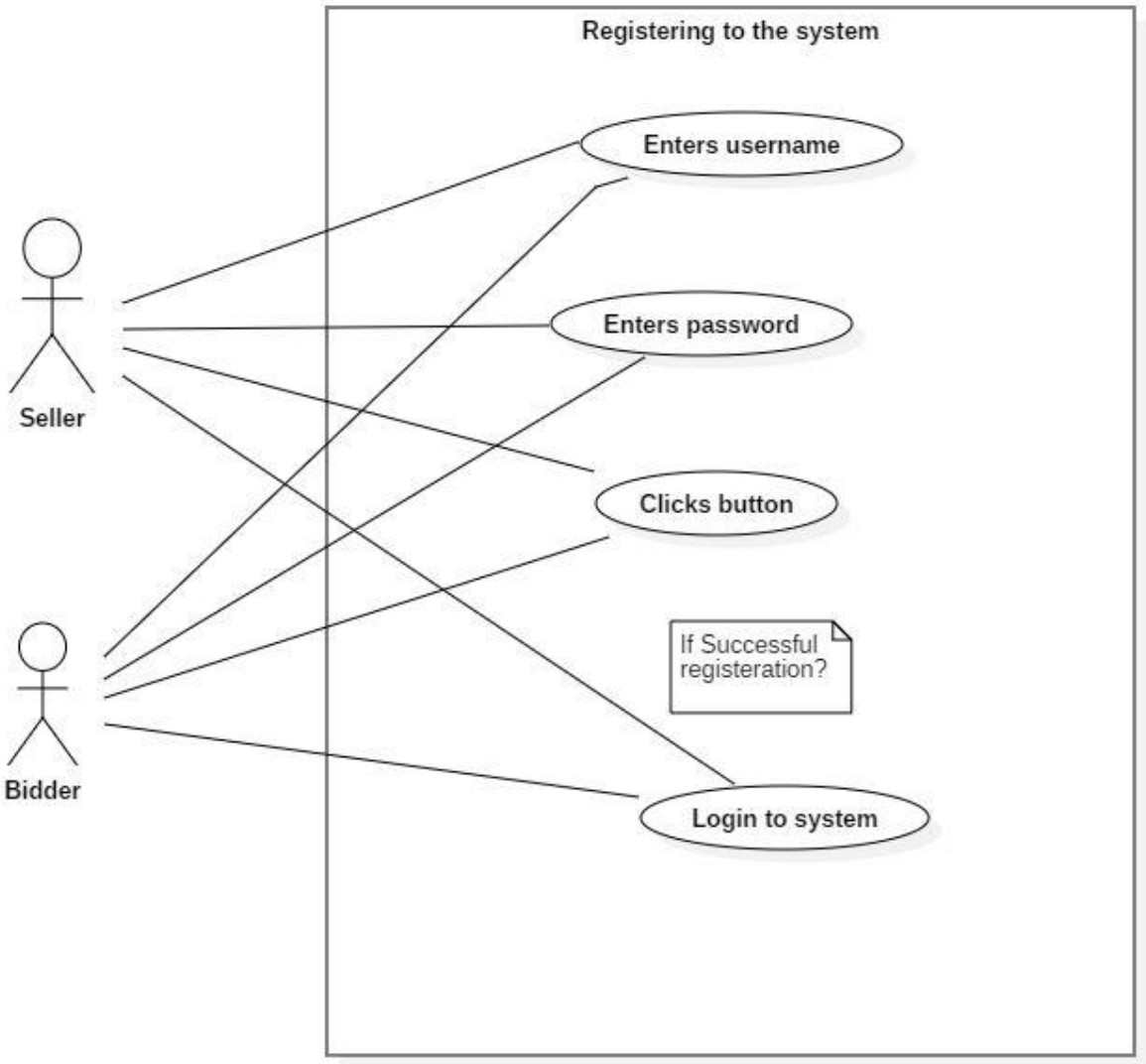
b) Operating System:

Windows 7/8/10+ (recommended), OS X 10.8+

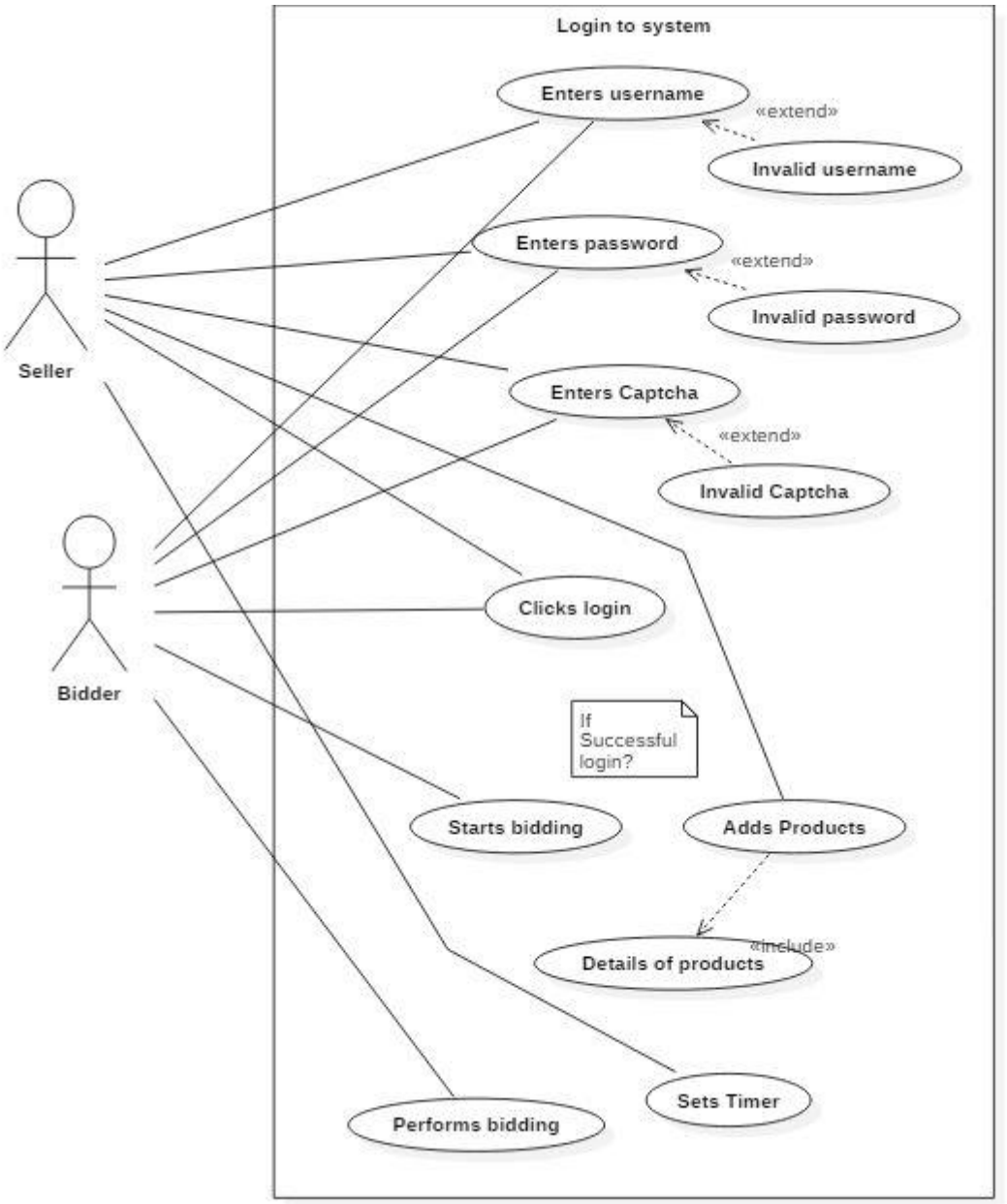
USECASE DIAGRAM

7. Use-case diagram

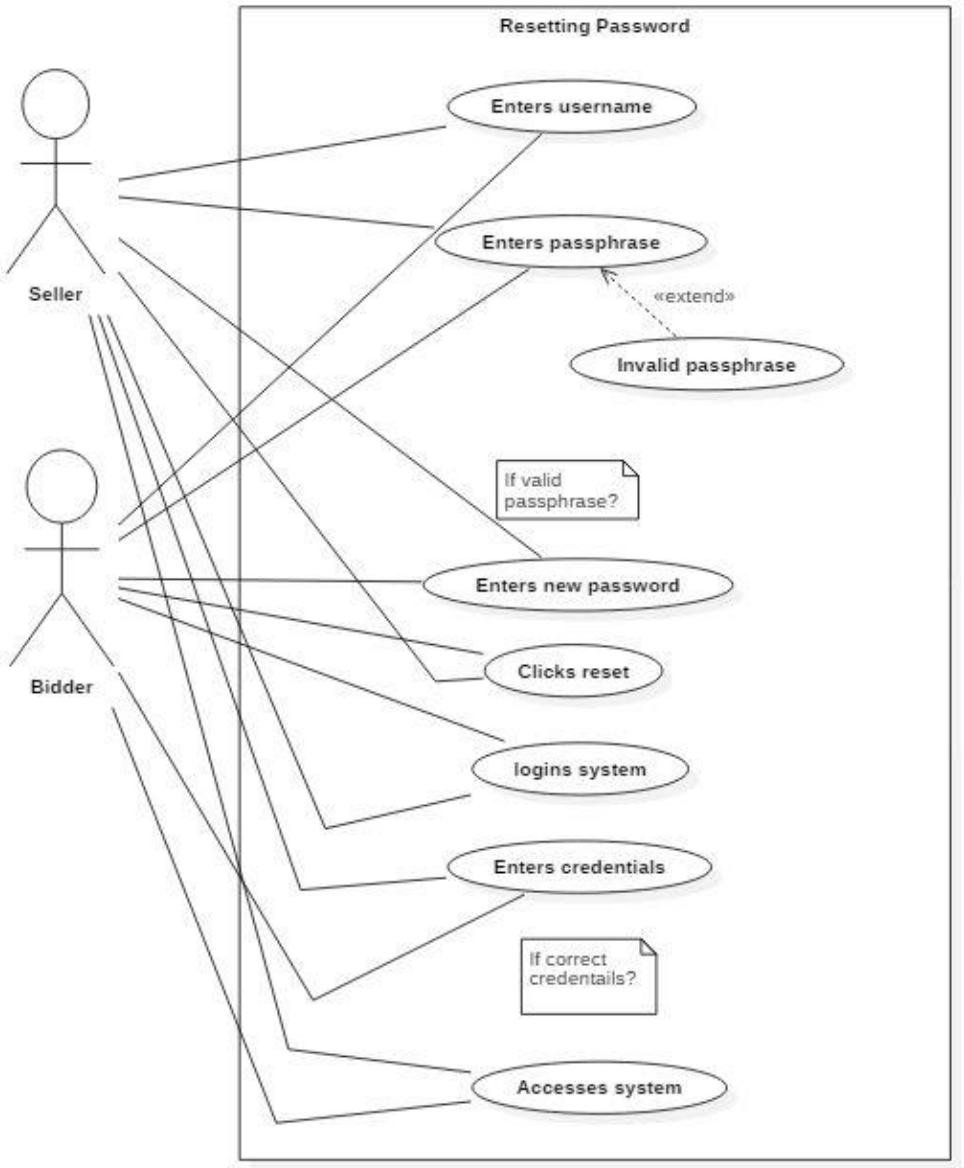
Scenario-1



Scenario -2

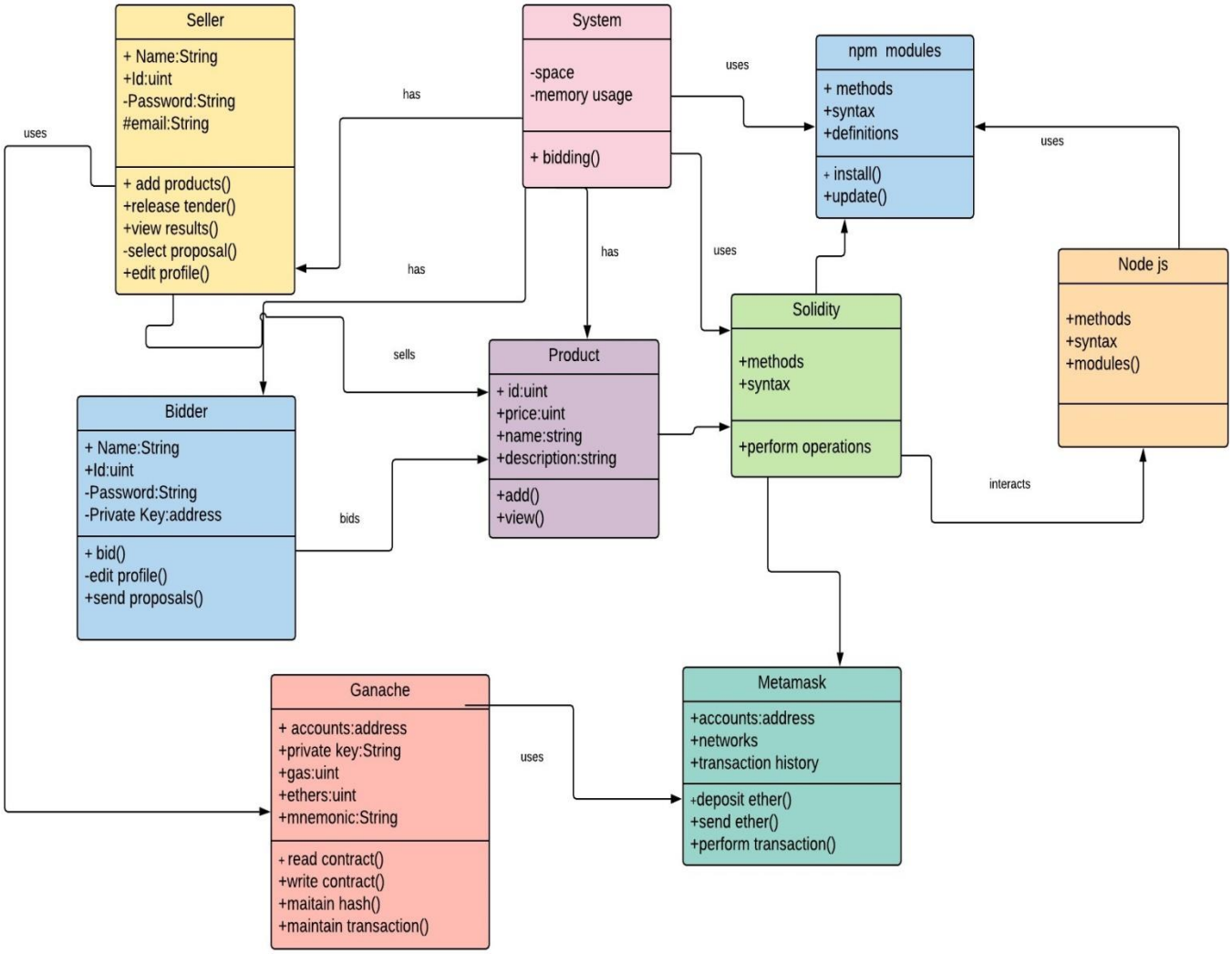

Login to system
Enters username
«extend»
Invalid username
Enters password
«extend»
Invalid password
Enters Captcha
«extend»
Invalid Captcha
Seller
Bidder
Clicks login
If Successful login?
Starts bidding
Adds Products
«include»
Details of products
Sets Timer
Performs bidding

Scenario-3


Resetting Password
Enters username
Enters passphrase
«extend»
Invalid passphrase
Seller
If valid passphrase?
Enters new password
Clicks reset
Bidder
logins system
Enters credentials
If correct credentails?
Accesses system

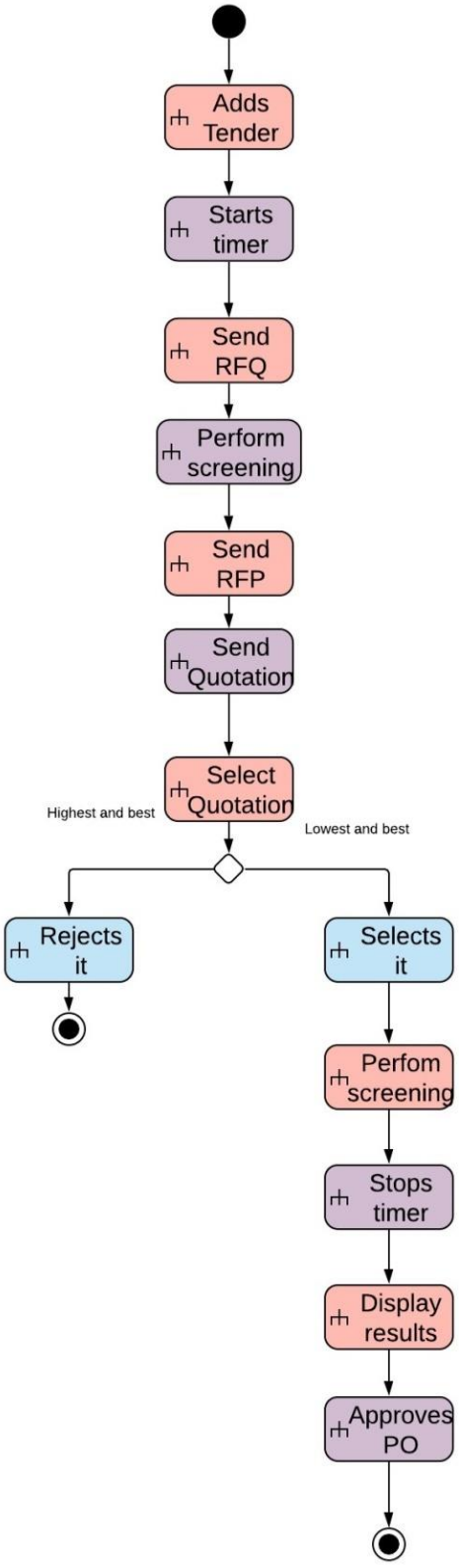
CLASS DIAGRAM

8. Class diagram



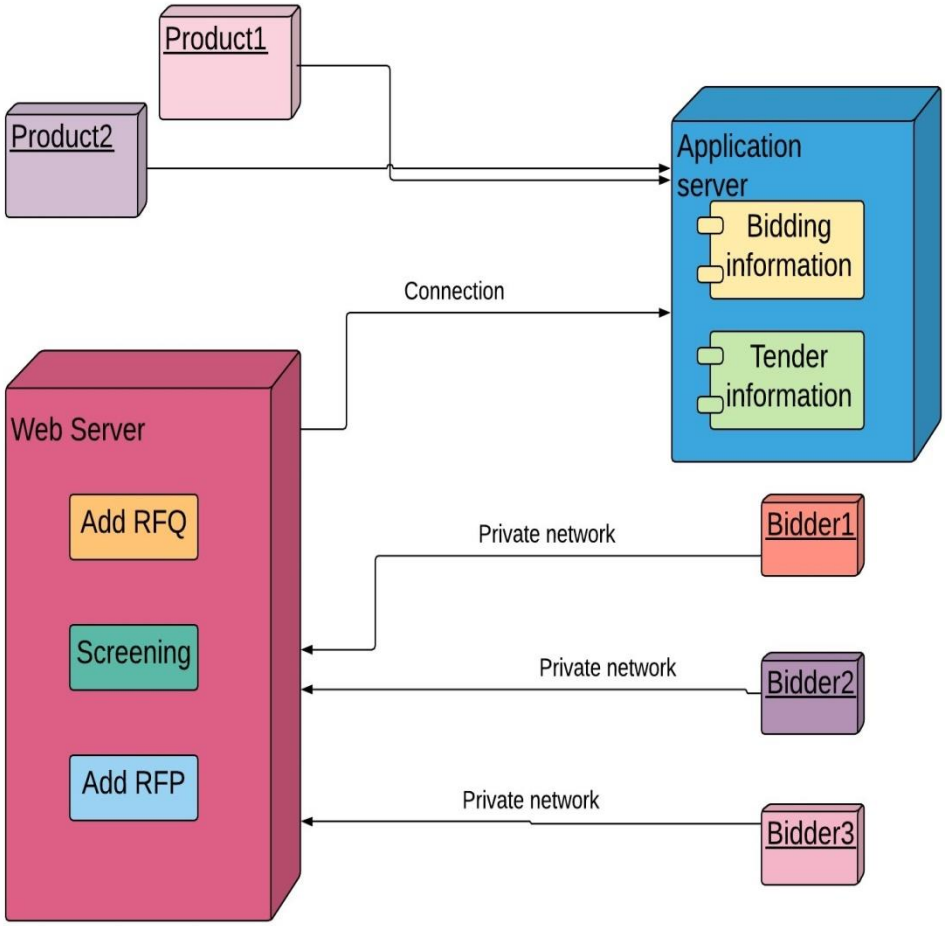
ACTIVITY DIAGRAM

9. Activity diagram



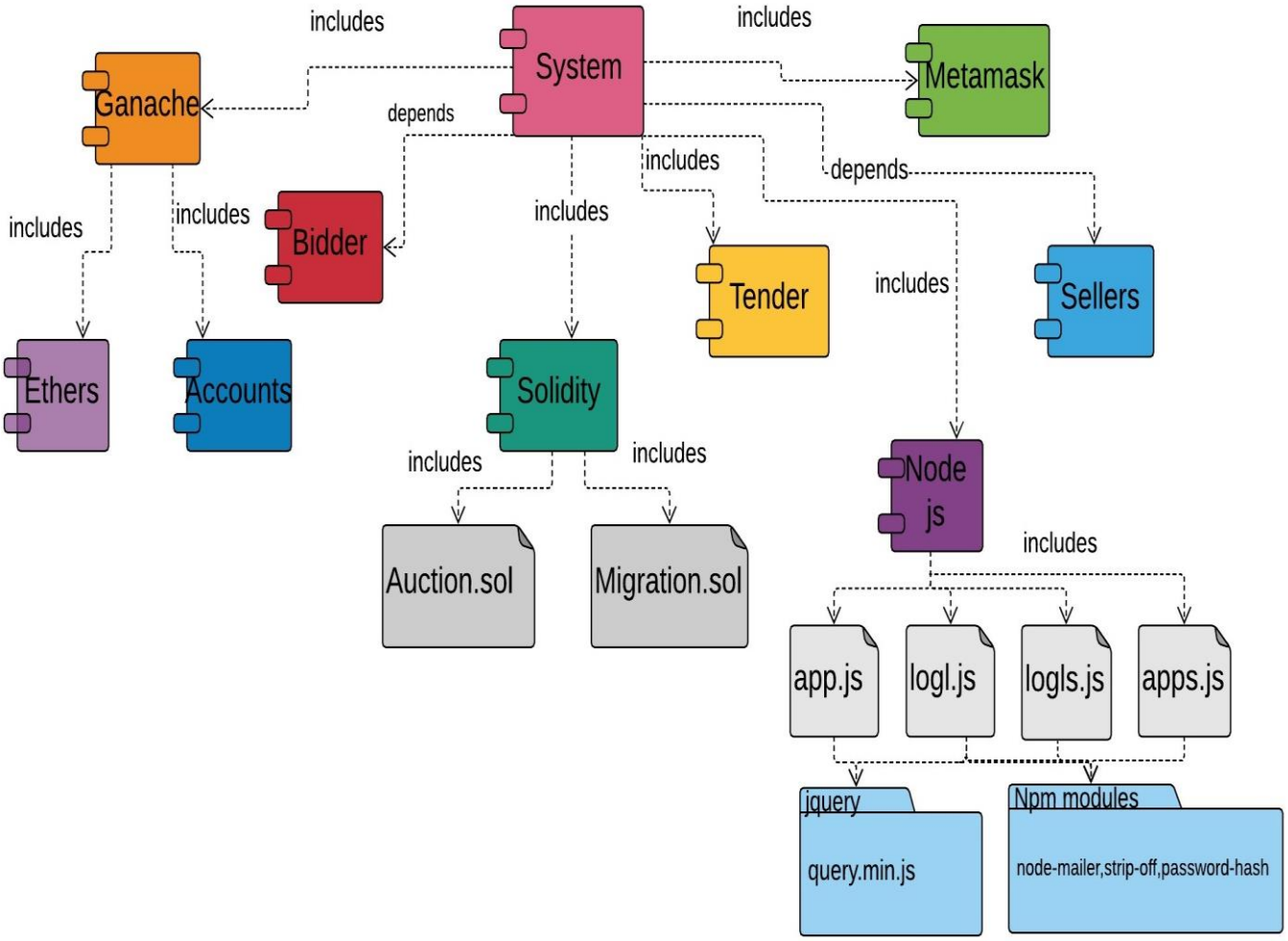
DEPLOYMENT DIAGRAM

10. Deployment diagram



COMPONENT DIAGRAM

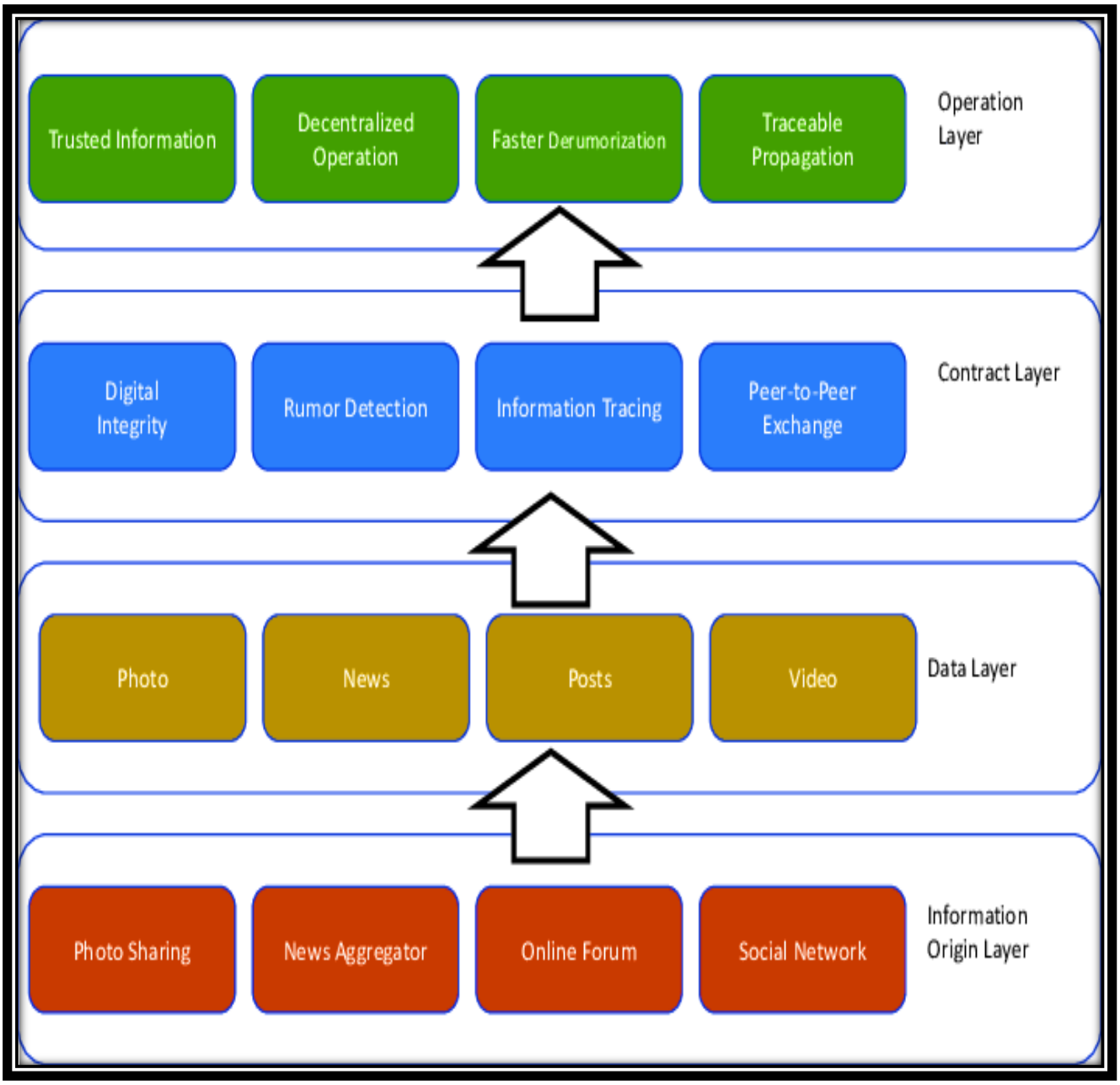
11. Component Diagram



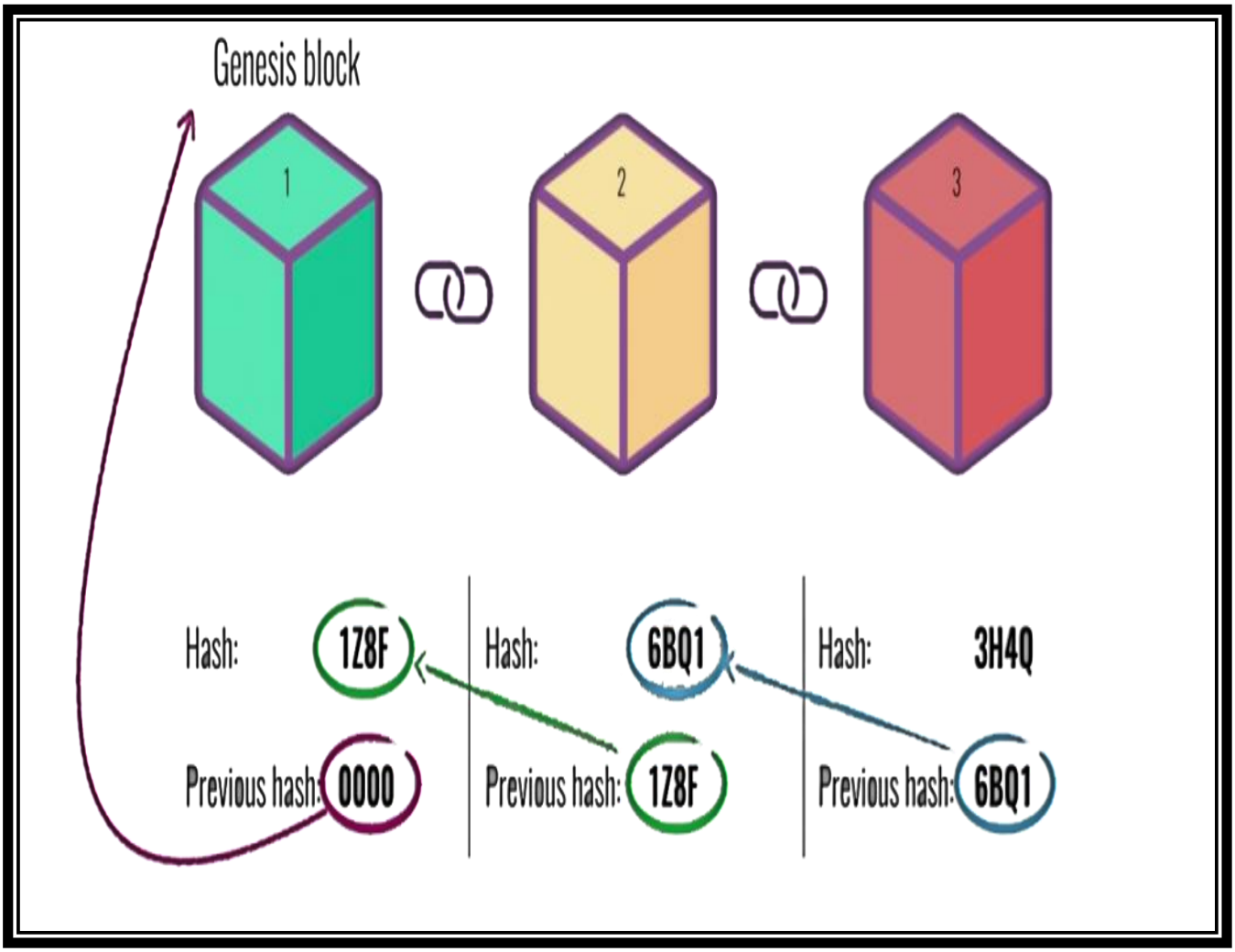
DIAGRAMS

12. Diagrams

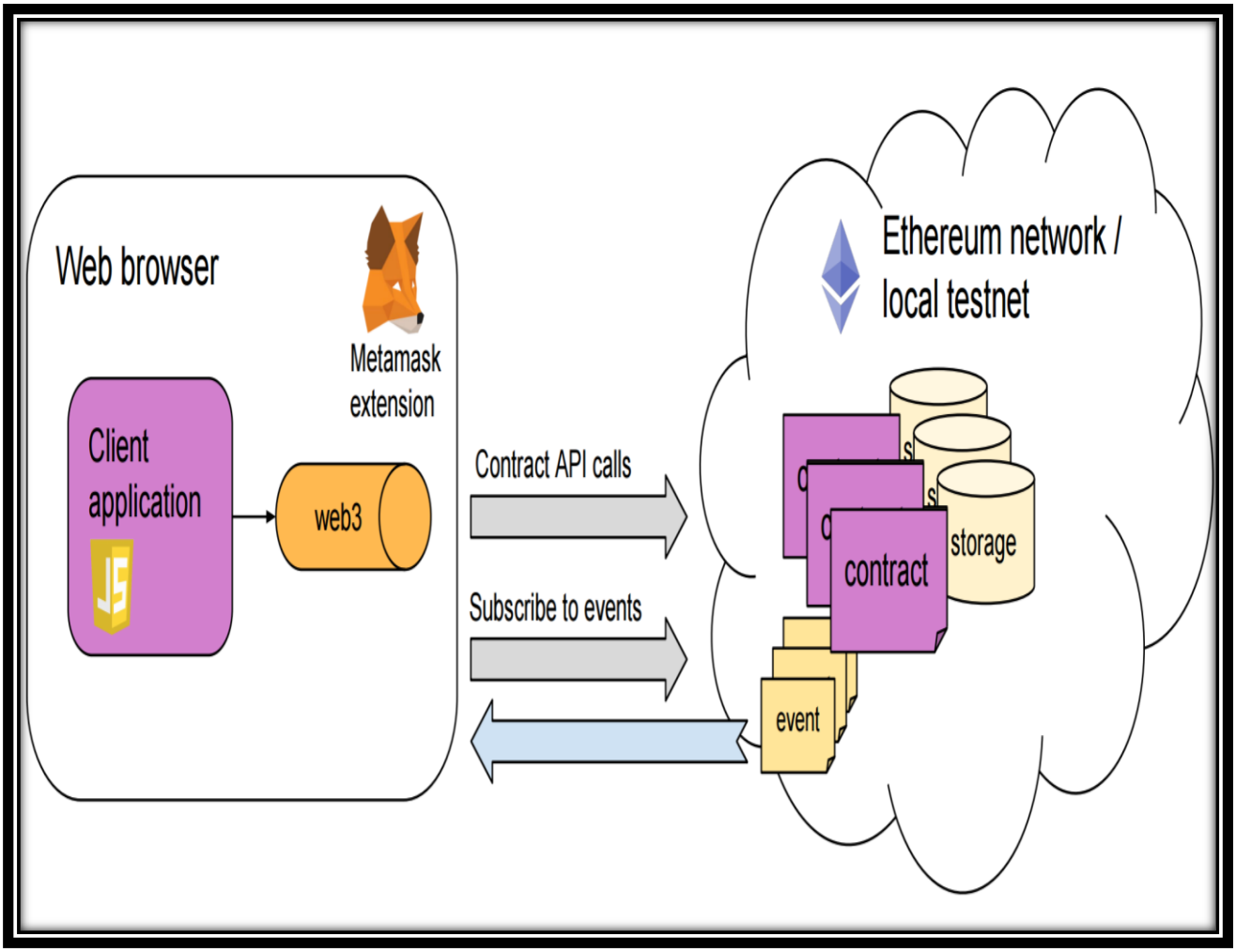
Blockchain Architecture



Block Diagram of Blockchain

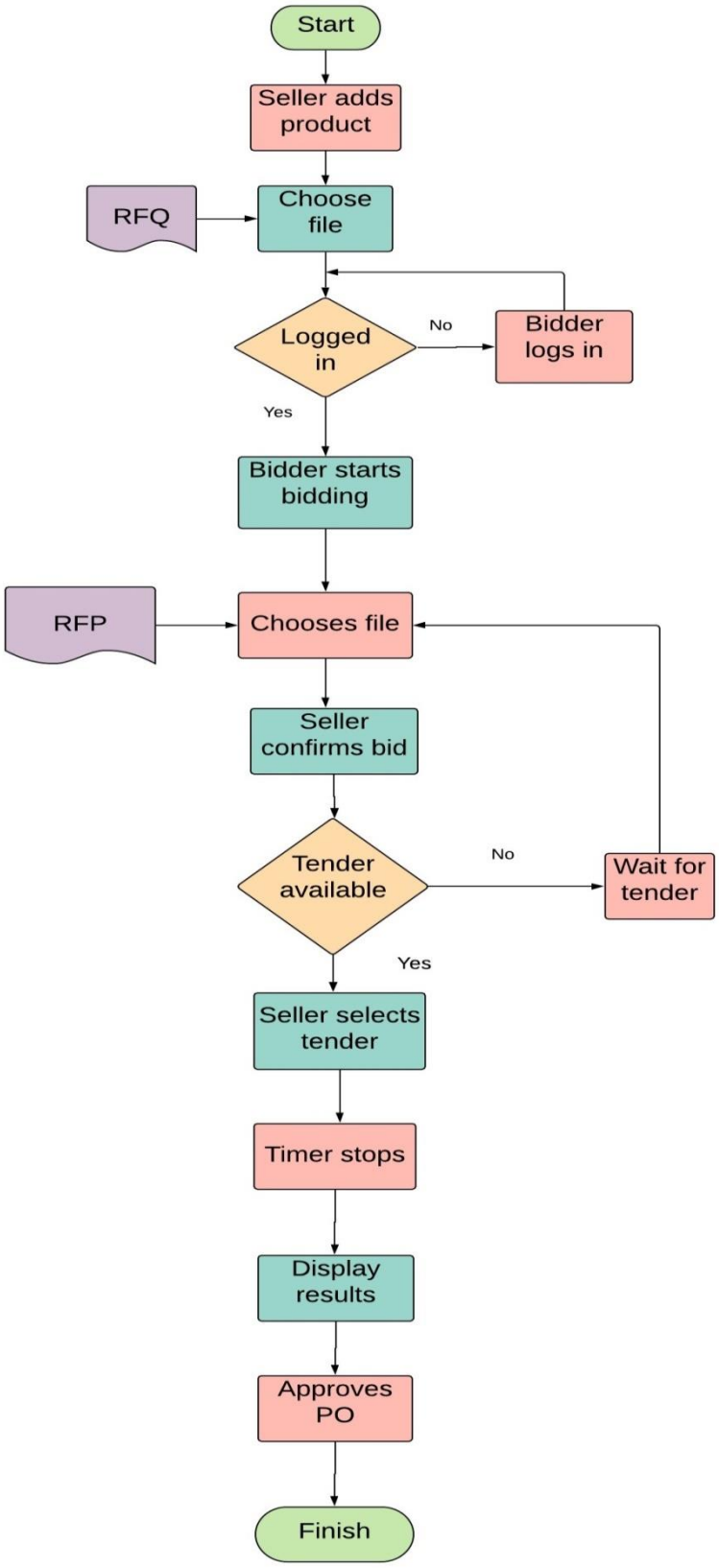


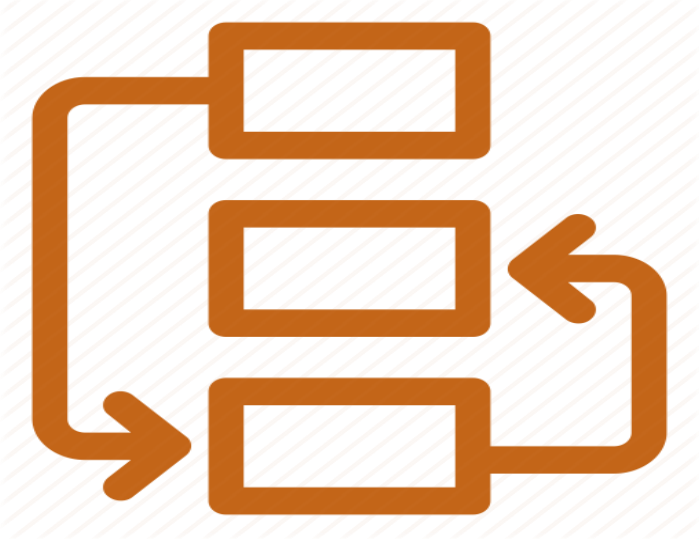
Interaction of Metamask and Ethereum network



FLOWCHART

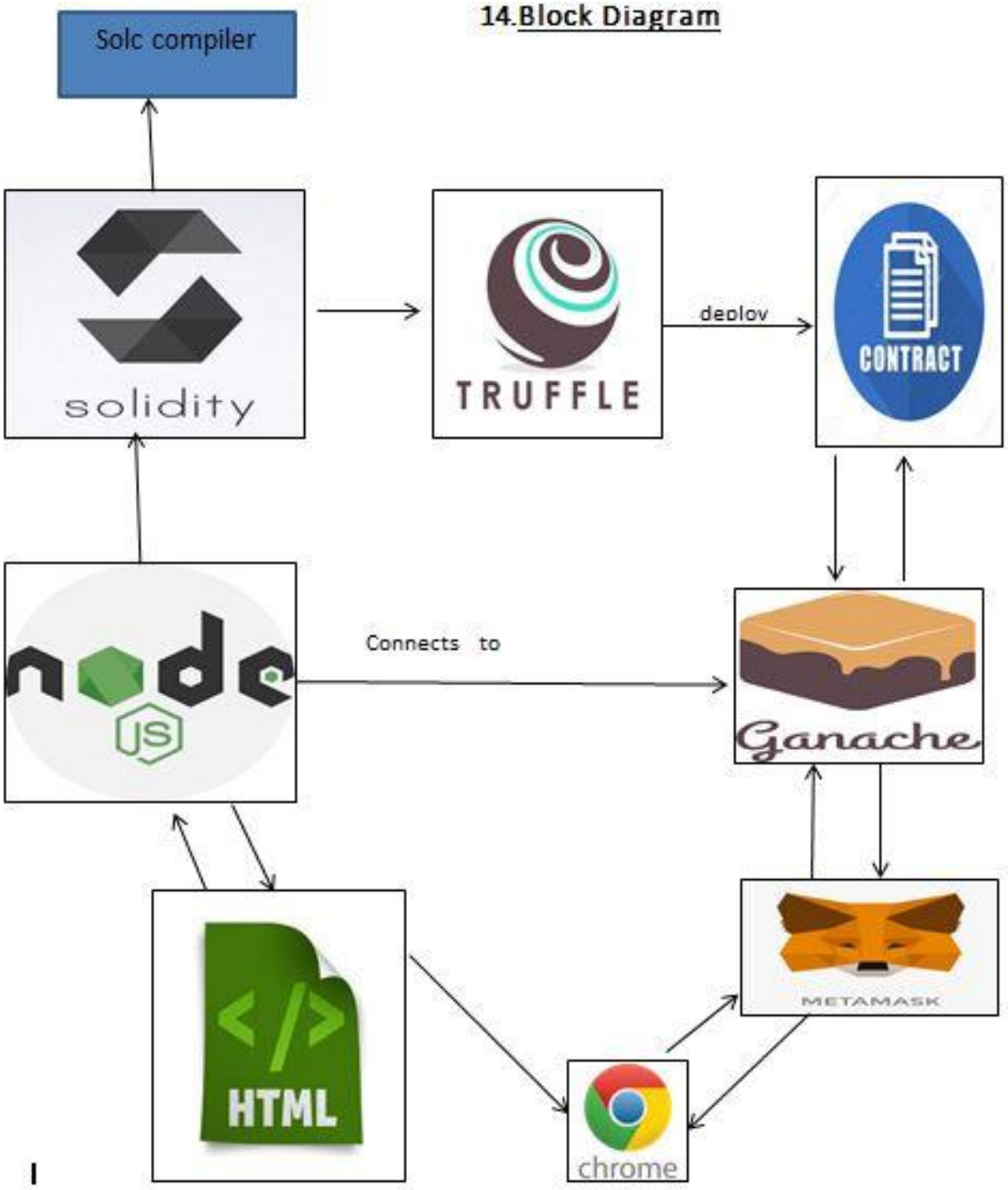
13. Flowchart





BLOCK DIAGRAM

14.Block Diagram



SCREEN-SHOTS

14. Screenshot

```
init: function() {
    return apps.initWeb3();
    },

    initWeb3: function() {
    // TODO: refactor conditional
    if (typeof web3 !== 'undefined') {
      // If a web3 instance is already provided by Meta Mask.
      apps.web3Provider = web3.currentProvider;
      web3 = new Web3(web3.currentProvider);
    } else {
      // Specify default instance if no web3 instance provided
      ethereum.enable();
      apps.web3Provider = new Web3.providers.HttpProvider('http://localhost:7545');
      web3 = new Web3(apps.web3Provider);
      Provider.enable();
      web3.enable();
      apps.web3Provider.enable();
    }
    return apps.initContract();
  },

  initContract: function() {

    $.getJSON("Auction.json", function(auction) {
      // Instantiate a new truffle contract from the artifact
      apps.contracts.Auction = TruffleContract(auction);
      // Connect provider to interact with contract
      apps.contracts.Auction.setProvider(apps.web3Provider);

      return apps.render();
    });
  },









App = {
    web3Provider: null,
    contracts: {},
```

```
  account: '0x0',
  hasVoted: false,
  docpath:null,
  sellid:0,
    init: function() {
    console.log("hfdj");
    return App.initWeb3();
  },


  initWeb3: function() {

    // TODO: refactor conditional
    if (typeof web3 !== 'undefined') {
      // If a web3 instance is already provided by Meta Mask.
      App.web3Provider = web3.currentProvider;
      web3 = new Web3(web3.currentProvider);
    } else {
      // Specify default instance if no web3 instance provided
      ethereum.enable();
      App.web3Provider = new Web3.providers.HttpProvider('http://localhost:7545');
      web3 = new Web3(App.web3Provider);
      Provider.enable();
      web3.enable();
      App.web3Provider.enable();
    }
    return App.initContract();
  },

  initContract: function() {
        $.getJSON("Auction.json", function(auction) {
      // Instantiate a new truffle contract from the artifact
      App.contracts.Auction = TruffleContract(auction);
      // Connect provider to interact with contract
      App.contracts.Auction.setProvider(App.web3Provider);

      return App.render();
    });
  },

  render: function() {

    var auctionInstance;
    //var loader = $("#loader");
    var content = $("#content");

    //loader.show();
    content.show();
```

```
    // Load account data
    web3.eth.getCoinbase(function(err, account) {
      if (err === null) {
        App.account = account;
        $("#accountAddress").html("Your Account: " + account);
      }
    });

    // Load contract data
    App.contracts.Auction.deployed().then(function(instance) {
      auctionInstance = instance;
      return auctionInstance.ProductCount();
    }).then(function(SellerCount) {
      if (SellerCount==0) {
        document.getElementById("bdbutton").style.display="none";
      }
      else
        {document.getElementById("addproduct").style.display="block";
      }
      var sellerResults = $("#sellerResults");
      sellerResults.empty();

      var sellerSelect = $('#sellerSelect');
      sellerSelect.empty();
var j=1;
function fun(mytimer,par,name){
var deadline = new Date(mytimer).getTime();
//console.log(par);
var x = setInterval(function() {

var now = new Date().getTime();
var t = deadline - now;
var days = Math.floor(t / (1000 * 60 * 60 * 24));
var hours = Math.floor((t%(1000 * 60 * 60 * 24))/(1000 * 60 * 60));
var minutes = Math.floor((t % (1000 * 60 * 60)) / (1000 * 60));
var seconds = Math.floor((t % (1000 * 60)) / 1000);
document.getElementById(par).innerHTML =days + "d " + hours + "h "
  + minutes + "m " + seconds + "s ";


if (t < 0) {
    clearInterval(x);
   // alert(x);
    document.getElementById(par).innerHTML ="   0 0  0 0";
  for (var f = 1; f <= SellerCount; f++) {
      auctionInstance.products(f).then(function(seller) {

          if(name==seller[1]&&seller[7]=="yes")
          {
```

```
            console.log("hello");
            // auctionInstance.addResult(seller[4],);
       // auctionInstance.updateProduct(seller[4],
{from:"0x6a5F9adc21Fa3747ceDBA229BD324EFcfC5635c8",gas:3000000 });
          }
          else
          {
            console.log("go");
          }

       });
     }

   }
  }, 1000);
  }

       for (var i = 1; i <= SellerCount; i++) {
         auctionInstance.products(i).then(function(seller) {
       //    j++;
       var there=seller[7];
       console.log(name);
            if(there=="no")
  {
    console.log("do nothing");
  }
  else{
          var id = seller[4];
          console.log(id);

          var name = seller[1];
          console.log(name);
          var amount = seller[3];
          var product = seller[2];

            var time=seller[5];
            var mg = seller[4]-1;
    var para1="color1 para"+j;

      var div1s=document.createElement('div');
    div1s.className="colt";


     document.getElementById('sellerResults').appendChild(div1s);


  var cont=document.createElement('div');
    cont.id="container  page-wrapper";
    cont.className="container  page-wrapper";
```

```
    div1s.appendChild(cont);


var pageinner=document.createElement('div');
  pageinner.id="page-inner";
  pageinner.className="page-inner";
   cont.appendChild(pageinner);




var roww=document.createElement('div');
  roww.id="row";
  roww.className="row";
   pageinner.appendChild(roww);




var wrapper=document.createElement('div');
  wrapper.id="el-wrapper";
  wrapper.className="el-wrapper";
   roww.appendChild(wrapper);


var boxup=document.createElement('div');
  boxup.id="box-up";
  boxup.className="box-up";
   wrapper.appendChild(boxup);



var img=document.createElement('img');
  img.id="img";
  img.className="img";
  img.src="js/images/pro"+mg+".png";
   boxup.appendChild(img);



var imginfo=document.createElement('div');
  imginfo.id="img-info";
  imginfo.className="img-info";
   boxup.appendChild(imginfo);



var infoinner=document.createElement('div');
  infoinner.id="info-inner";
```

```
  infoinner.className="info-inner";
   imginfo.appendChild(infoinner);



var span=document.createElement('span');
  span.id="p-company";
  span.className="p-company";
    span.innerHTML=name;
   infoinner.appendChild(span);


var span1=document.createElement('span');
  span1.id="p-name";
  span1.className="p-name";
    span1.innerHTML="name : "+ name;
   infoinner.appendChild(span1);


var asize=document.createElement('div');
  asize.id="a-size";
  asize.className="a-size";
   imginfo.appendChild(asize);


   var p1=document.createElement('h2');
  p1.id=para1;
  p1.className=para1;
  asize.appendChild(p1);
  fun(time,para1,name);
  var boxdown=document.createElement('div');
  boxdown.id="box-down";
  boxdown.className="box-down";
  wrapper.appendChild(boxdown);

  var hbg=document.createElement('div');
  hbg.id="h-bg";
  hbg.className="h-bg";
   boxdown.appendChild(hbg);

  var hbginner=document.createElement('div');
  hbginner.id="h-bg-inner";
  hbginner.className="h-bg-inner";
  hbg.appendChild(hbginner);

  var a=document.createElement('a');
  a.id="cart";
  a.href="#";
```

```
    a.className="cart";
    boxdown.appendChild(a);

    var spanprice=document.createElement('span');
    spanprice.id="price";
    spanprice.className="price";
    spanprice.innerHTML="$"+amount;
    a.appendChild(spanprice);

    var addtocart=document.createElement('span');
    addtocart.id="add-to-cart";
    addtocart.className="add-to-cart";
    a.appendChild(addtocart);

    var txt=document.createElement('span');
    txt.id="txt";
    txt.className="txt";
    txt.innerHTML=name;
    addtocart.appendChild(txt);

    var sellerOption = "<option value='" + id + "' >" + name + "</ option>";
            sellerSelect.append(sellerOption);

    j++;
    $("#content").show();


    }
});} })
} ,

bidconf:function()
 {

       App.contracts.Auction.deployed().then(function(instance) {
       auctionInstance = instance;

       return auctionInstance.ResultCount();
       }).then(function(ResultCount) {

       if (ResultCount==0) {
       Console.log("do nothing")
       }
        else
        {
               sessionStorage.setItem("there","no");
               sessionStorage.setItem("productkaCount",0);
               for (var i = 1; i <= ResultCount; i++) {
               auctionInstance.results(i).then(function(result) {
```

```
                if(result[0]==sessionStorage.getItem('loggedsellerid')&&result[8]=="yes")
                {
                                        sessionStorage.setItem("there","yes");
                                        var sellerId=result[0];
                                        var prodname=result[1];
                                        var proddesc=result[2];
                                        var prodId=result[4];
                                        var prodTime=result[5];
                                        var resId=result[6];
                                        var Count=sessionStorage.getItem("productkaCount");
                                        Count++;
                                        sessionStorage.setItem("productkaCount",Count);
                                        var
productkaid="productkaid"+sessionStorage.getItem("productkaCount");
                                        sessionStorage.setItem(productkaid,prodId);
                                        var
productkaname="productkaname"+sessionStorage.getItem("productkaCount");
                                        sessionStorage.setItem(productkaname,prodname);
                                        var
res="resId"+sessionStorage.getItem("productkaCount");
                                        sessionStorage.setItem(res,resId);

                                         }))}}
                                        return
auctionInstance.BidCount();}).then(function(BidCount){
                        if(sessionStorage.getItem("there")=="yes")
                        {
                            console.log("b8");
                            sessionStorage.setItem("procount",1);
                            for(var j=1;j<=sessionStorage.getItem("productkaCount");j++)
                            {
                                    var productkaid="productkaid"+j;
                                    var productkaname="productkaname"+j;
                                    var content=document.createElement('div');
                                    //gui part start*******************
                                    content.id="content";
                                    content.className="content";
                                    content.innerHTML=sessionStorage.getItem(productkaid);
                                    document.getElementById('bdcf').appendChild(content);

                                    var tablename="table"+j;
                                    var table=document.createElement('table');
                                    table.id=tablename;
                                    table.className=tablename;
                                    content.appendChild(table);
                                    // gui part end*****************
                             var l=0;
                            for (var i = 1; i <= BidCount; i++) {
                                    var productkaid=0;
```

```
                                    var resId=0;
                                    var res=0;
                                    auctionInstance.bids(i).then(function(result) {
                                    if(l==BidCount)
                                    {
                                        l=1;
                                    }
                                    else
                                    {
                                         l++;
                                    }

                                if(l==1)
                                {
                                productkaid="productkaid"+sessionStorage.getItem("procount");
                                resId="resId"+sessionStorage.getItem("procount");
                                var add=sessionStorage.getItem("procount");
                                add++;
                                sessionStorage.setItem("procount",add)
                                }
                                res=sessionStorage.getItem(resId);
if(result[0]==sessionStorage.getItem(productkaid)&&result[1]==sessionStorage.getItem
          ("loggedsellerid"))
{
  var bidId=result[2];
  var prodamt=result[3];
  var filename=result[4];
  var count=sessionStorage.getItem("procount");
  count--;
  tablee="table"+count;

  var tr=document.createElement("tr");
  tr.id="tr1";
  tr.className="tr1";
  document.getElementById(tablee).appendChild(tr);



  var td1=document.createElement("td");
  td1.id="td2";
  td1.className="td2";
  td1.innerHTML=bidId;
  tr.appendChild(td1);



  var td2=document.createElement("td");
  td2.id="td2";
  td2.className="td2";
```

```
    td2.innerHTML="buyer id";
    tr.appendChild(td2);


    var td3=document.createElement("td");
    td3.id="td3";
    td3.className="td3";
    td3.innerHTML=prodamt;
    tr.appendChild(td3);



    var td4=document.createElement("td");
    td4.id="td4";
    td4.className="td4";
    tr.appendChild(td4);

    var aa1 = document.createElement('a');
    var linkText = document.createTextNode("View RFQ");
    aa1.appendChild(linkText);
    aa1.title = "my title text";
    aa1.href = "#";
    aa1.id="txt";
    aa1.className="txt";
    td4.appendChild(aa1);
    aa1.onclick=function()
    {
        window.open('js/docsbuyer/'+filename, '_blank', 'fullscreen=yes');
        return false;
    }

    var td5=document.createElement("td");
    td5.id="td3";
    td5.className="td3";

    tr.appendChild(td5);
    var but=document.createElement("button");
    but.id="butt"+l;
    but.className="butt"+l;
    but.value=bidId;
    but.onclick = function(){ App.confirmit(); };
    td5.appendChild(but);

    var text=document.createElement("p");
    text.id="text";
    text.className="text";
    text.innerHTML="bd now";
    document.getElementById("butt"+l).appendChild(text);
    var input = document.createElement("input");
```

```
    input.setAttribute("type", "hidden");
    input.setAttribute("id", "hd");
    input.setAttribute("value", res);

  //append to form element that you want .
  document.getElementById("bdcf").appendChild(input);

}
})}}
}
else
{
    console.log("DO NOTHING")
}

    })
},
  confirmit:function()
  {  var buttom=document.getElementById("butt1").value;
     var res=document.getElementById("hd").value;
     alert(buttom);
     alert(res);
     App.contracts.Auction.deployed().then(function(instance)
     {
        auctionInstance=instance;
        return auctionInstance.addFinalResult(res,buttom,{
from:"0x1a2a05C4525c0B7497164b9BF8023F0269d0D39b",gas:3000000 });
     })
  },
  result:function()
  {
    App.contracts.Auction.deployed().then(function(instance) {
    auctionInstance = instance;
    return auctionInstance.FinalResultCount();
  }).then(function(ResultCount) {

    if (ResultCount==0)
    {
        console.log("Do nothing");
    }
    else
    {

        for (var i = 1; i <= ResultCount; i++)
        {
            auctionInstance.finalresults(i).then(function(result) {
            var sellerId=result[0];
            var prodname=result[1];
            var proddesc=result[2];
```

```
            var prodId=result[3];
            var prodTime=result[4];
            var resId=result[5];
            var bidId=result[6];
            var FinalResultId=result[7];

    var div1s=document.createElement('div');
    div1s.className="colt";
    document.getElementById('result').appendChild(div1s);

    var cont=document.createElement('div');
    cont.id="container  page-wrapper";
    cont.className="container  page-wrapper";
    div1s.appendChild(cont);

    var pageinner=document.createElement('div');
    pageinner.id="page-inner";
    pageinner.className="page-inner";
    cont.appendChild(pageinner);

    var roww=document.createElement('div');
    roww.id="row";
    roww.className="row";
    pageinner.appendChild(roww);

    var wrapper=document.createElement('div');
    wrapper.id="el-wrapper";
    wrapper.className="el-wrapper";
    roww.appendChild(wrapper);

    var boxup=document.createElement('div');
    boxup.id="box-up";
    boxup.className="box-up";
    wrapper.appendChild(boxup);

    var img=document.createElement('img');
    img.id="img";
    img.className="img";
    img.src="http://code.slicecrowd.com/labs/4/images/t-shirt.png";
    boxup.appendChild(img);

    var imginfo=document.createElement('div');
    imginfo.id="img-info";
    imginfo.className="img-info";
    boxup.appendChild(imginfo);

    var infoinner=document.createElement('div');
    infoinner.id="info-inner";
    infoinner.className="info-inner";
```

```
    imginfo.appendChild(infoinner);

    var span=document.createElement('span');
    span.id="p-company";
    span.className="p-company";
    span.innerHTML=prodname;
    infoinner.appendChild(span);

    var span1=document.createElement('span');
    span1.id="p-name";
    span1.className="p-name";
    span1.innerHTML="resid : "+ resId;
    infoinner.appendChild(span1);

    var asize=document.createElement('div');
    asize.id="a-size";
    asize.className="a-size";
    imginfo.appendChild(asize);

    var p1=document.createElement('h2');
    p1.id="para";
    p1.className="para";
    p1.innerHTML="prodid:"+prodId;
    asize.appendChild(p1);

    var boxdown=document.createElement('div');
    boxdown.id="box-down";
    boxdown.className="box-down";
    wrapper.appendChild(boxdown);

    var hbg=document.createElement('div');
    hbg.id="h-bg";
    hbg.className="h-bg";
    boxdown.appendChild(hbg);

    var hbginner=document.createElement('div');
    hbginner.id="h-bg-inner";
    hbginner.className="h-bg-inner";
    hbg.appendChild(hbginner);

    var a=document.createElement('a');
    a.id="cart";
    a.href="#";
    a.className="cart";
    boxdown.appendChild(a);

    var spanprice=document.createElement('span');
    spanprice.id="price";
    spanprice.className="price";
```

```
    spanprice.innerHTML="bid id:"+bidId;
    a.appendChild(spanprice);

    var addtocart=document.createElement('span');
    addtocart.id="add-to-cart";
    addtocart.className="add-to-cart";
    a.appendChild(addtocart);

    var txt=document.createElement('span');
    txt.id="txt";
    txt.className="txt";
    txt.innerHTML="name"+FinalResultId;
    addtocart.appendChild(txt);
    })
}}})

},
    bidNow: function()
    {
        var prodId = $('#sellerSelect').val();
        sessionStorage.setItem("prodId",prodId);
        App.contracts.Auction.deployed().then(function(instance) {
        auctionInstance = instance;
        return auctionInstance.ProductCount();
    }).then(function(ProductCount) {
        var j=1;
        for (var i = 1; i <= ProductCount; i++)
        {
            auctionInstance.products(i).then(function(sellers) {
            if(sellers[4]==prodId)
            {
                App.sel=sellers[0];
                alert("App.sellid");
                alert(App.sel);
            }

                 j++;
        })
    }
        return auctionInstance.BidCount();
    }).then(function(bidCount)
    {
        bidCount++;

App.docpath="C:/auct1/auct/src/js/docsbuyer/pro"+prodId+"sell"+App.sel+"bid"+bidCount+".pd
f";
        alert(App.docpath);

sessionStorage.setItem("filename","pro"+prodId+"sell"+App.sel+"bid"+bidCount+".pdf");
```

```
        window.location.assign("http://localhost:8089?docpath="+App.docpath);
    })
    return false;
    }
};

  $(function() {
    $(window).load(function() {
    App.init();
    });
  });
```

```
addprod: function()
{
    var prodname =document.getElementById('pn');
```

```
    var proddesc =document.getElementById('pd');
    var prodprice =document.getElementById('amount');
    var time =document.getElementById('timee');
    prod.contracts.Auction.deployed().then(function(instance) {
    auctionInstance = instance;
    return auctionInstance.ProductCount();}).then(function(ProductCount)
    {
        var account= "0x1a2a05C4525c0B7497164b9BF8023F0269d0D39b";
        alert("n logl.flag for");
        var id=sessionStorage.getItem("loggedsellerid");
        alert(ProductCount);
        ProductCount++;
        alert("ProductCount"+ProductCount);
        prod.path="C:/auct1/auct/src/js/images/pro"+ProductCount+".png";
        prod.docpath="C:/auct1/auct/src/js/docs/pro"+ProductCount+".pdf";
        return
auctionInstance.addProduct(id,prodname.value,proddesc.value,prodprice.value,time.value,pro
d.path, { from: account,gas:3000000 });
        }).then(function(result)
        {

window.location.assign("http://localhost:8086?path="+prod.path+"&docpath="+prod.docpath);
        }).catch(function(error){
        console.log(error);
    });
}
},
```

```
logl =
{
  web3:null,
  i:0,
  ref:0,
  privateKey:null,
  web3Provider: null,
  contracts: {},
  account: '0x0',
  hasVoted: false,
  flag:0,
  init: function() {

    return logl.initWeb3();
  },

  login: function() {
  if(window.event.target.value=="login")
  {
    var string1 = removeSpaces(document.getElementById('mainCaptcha').value);
    var string2 = removeSpaces(document.getElementById('txtInput').value);
    if (string1 == string2)
    {

    }
    else
    {
      alert("enter Captcha correctly");
      window.location.assign("indexb.html");
    }

    function removeSpaces(string)
    {
      return string.split(' ').join('');
    }

    var username =document.getElementById('user').value;
    var password =document.getElementById('pass').value;
    var flag2=0;
      logl.contracts.Auction.deployed().then(function(instance)
      {
        auctionInstance = instance;
        return auctionInstance.BuyerCount();
      }).then(function(BuyerCount) {
      for (var i = 1; i <= BuyerCount; i++)
      {
        auctionInstance.bidder(i).then(function(bidders) {
        logl.ref++;
        var name = bidders[1];
```

```
        var id=bidders[0];
        var pass = bidders[3];
        var acc = bidders[2];
        var hash=bidders[5];
        var salt=bidders[4];
        var crypto = require('crypto');
        var genRandomString = function(length){
        return crypto.randomBytes(Math.ceil(length/2))
        .toString('hex') /** convert to hexadecimal format */
        .slice(0,length);   /** return required number of characters */
  };

  var sha512 = function(password, salt)
  {
  var hash = crypto.createHmac('sha512', salt); /** Hashing algorithm sha512 */
  hash.update(password);
  var value = hash.digest('hex');
  return {salt:salt,
          passwordHash:value
          };
  };
  var data = sha512(password, salt);
  if (data.passwordHash==hash && name==username)
  {
    sessionStorage.setItem("loggedseller", name);
    sessionStorage.setItem("loggedsellerid",id );
    sessionStorage.setItem("seller","0" );
    sessionStorage.setItem("buyer","1" );
    sessionStorage.setItem("account",acc );
    flag2=1;
    window.location.assign("index.html");


  }
  else
  {
    if(logl.ref==BuyerCount)
    {
      if(flag2==1)
      {
        alert("Welcome");
        window.location.assign("index.html");}
      else
      {
        alert("enter correctly");
        window.location.assign("indexb.html");
      }}
  }
});
```

```
}
  })
}
  else if(window.event.target.value=="register")
  {
      var verbs, nouns, adjectives, adverbs, preposition;
      nouns = ["bird", "clock", "boy", "plastic", "duck", "teacher", "old lady",
"professor", "hamster", "dog"];
      verbs = ["kicked", "ran", "flew", "dodged", "sliced", "rolled", "died", "breathed",
"slept", "killed"];
      adjectives = ["beautiful", "lazy", "professional", "lovely", "dumb", "rough",
"soft", "hot", "vibrating", "slimy"];
      adverbs = ["slowly", "elegantly", "precisely", "quickly", "sadly", "humbly",
"proudly", "shockingly", "calmly", "passionately"];
      preposition = ["down", "into", "up", "on", "upon", "below", "above", "through",
"across", "towards"];
      function sentence()
      {
          var rand1 = Math.floor(Math.random() * 10);
          var rand2 = Math.floor(Math.random() * 10);
          var rand3 = Math.floor(Math.random() * 10);
          var rand4 = Math.floor(Math.random() * 10);
          var rand5 = Math.floor(Math.random() * 10);
          var rand6 = Math.floor(Math.random() * 10);
          var content = "The " + adjectives[rand1] + " " + nouns[rand2] + " " +
adverbs[rand3] + " " + verbs[rand4] + " because some " + nouns[rand1] + " " +
adverbs[rand1] + " " + verbs[rand1] + " " + preposition[rand1] + " a " + adjectives[rand2]
+ " " + nouns[rand5] + " which, became a " + adjectives[rand3] + " " + adjectives[rand4] +
" " + nouns[rand6];
          return content;
      };

  var SN = require('sync-node');
  var pn = SN.createQueue();
  pn.pushJob(function()
  {
    return new Promise(function (resolve, reject) {
    setTimeout(function()
    {
      var username =document.getElementById('userr');
      logl.contracts.Auction.deployed().then(function(instance)
      {
        auctionInstance = instance;
        return auctionInstance.BuyerCount();}).then(function(BuyerCount){
        var valueToBeReturned = 0;
        if (BuyerCount==0) {resolve(valueToBeReturned);
      }
      for ( var  j = 1; j <= BuyerCount; j++)
      {
```

```
        auctionInstance.bidder(j).then(function(bidders)
        {
          logl.i++;
          var name = bidders[1];
          if (name==username.value)
          {
             valueToBeReturned=1;
          if (logl.i==BuyerCount)
          {
            resolve(valueToBeReturned);

          }

      }
        else
        {


if (logl.i==BuyerCount) {
              resolve(valueToBeReturned);

            }


            }

            });}

    })


        }, 5000);

    })
});
    pn.pushJob(function(res){
    setTimeout(function()
      {if(res==0){

        logl.contracts.Auction.deployed().then(function(instance) {
      auctionInstance = instance;
      return auctionInstance.BuyerCount();}).then(function(BuyerCount) {
        var password =document.getElementById('passr').value;

         var crypto = require('crypto');
var genRandomString = function(length){
    return crypto.randomBytes(Math.ceil(length/2))
            .toString('hex') /** convert to hexadecimal format */
            .slice(0,length);   /** return required number of characters */
};
```

```
var sha512 = function(password, salt){
    var hash = crypto.createHmac('sha512', salt); /** Hashing algorithm sha512 */
    hash.update(password);
    var value = hash.digest('hex');
    return {
        salt:salt,
        passwordHash:value
    };
};
  var passwordData;
function saltHashPassword(userpassword) {
    var salt = genRandomString(16); /** Gives us salt of length 16 */
    passwordData = sha512(userpassword, salt);
   // return new Promise(function(fullfil,reject))


 }

saltHashPassword(password);



    var username =document.getElementById('userr');


       const bip39 = require('bip39');
     const hdkey = require('ethereumjs-wallet/hdkey');
  const mnemonic = "wool seminar turkey hedgehog table fat error adult truck hood voyage
clown";
   const hdPath = "m/44'/60'/0'/0/0";
      var index=BuyerCount;
    const hdwallet = hdkey.fromMasterSeed(bip39.mnemonicToSeed(mnemonic));
    const node = hdwallet.derivePath(hdPath + String(index));
   const stripHexPrefix = require('strip-hex-prefix');
   var passphrase=sentence();


   logl.account= node.getWallet().getAddressString(),
    logl.privateKey= node.getWallet().getPrivateKeyString(),
   logl.privateKey=stripHexPrefix(logl.privateKey);



  return
auctionInstance.addBidder(username.value,password,passwordData.salt,passwordData.passwordH
ash,logl.account,passphrase,logl.privateKey, { from: logl.account,gas:3000000 });

}).then(function(result){
  alert("successfully reg plss log");
```

```
  window.location.assign("indexb.html");
}).catch(function(error));

}
else
{

  alert("retry");
  window.location.assign("indexb.html");
}
   },5000);

   });
   }
 //
 else if(window.event.target.value=="reset")
    {var user =document.getElementById('user2');
    var passphrase =document.getElementById('passphrase2');
//fetching data from contract
    logl.contracts.Auction.deployed().then(function(instance) {
      auctionInstance = instance;
      return auctionInstance.BuyerCount();
    }).then(function(BuyerCount) {
      var flag1=0;
      var flag2=0;
      if(BuyerCount==0)
      {
        alert("no such account det exists");
        window.location.assign("indexb.html");
      }
      for (var i = 1; i <= BuyerCount; i++) {

        auctionInstance.bidder(i).then(function(bidders) {
        //  var id = bidders[0];
          var name = bidders[1];
         // var add = bidders[2];
          //var pass = bidders[3];
           var passphra = bidders[4];
           flag1++;

           if (user.value==name && passphrase.value==passphra)
           {
            alert("welcome");
            flag1=1;
flag2=1;
           window.location.assign("index.html");
           }
           else
           {
```

```
            if(flag1==BuyerCount)
            {if(flag2==0)
              {alert("no such credentials found");
              window.location.assign("indexb.html");}
              else
                          { window.location.assign("index.html");
           }

            }
            }

                        });
      }


    })
}
}
   },
  $(function() {
  $(window).load(function() {
    logl.init();
  });
});
```

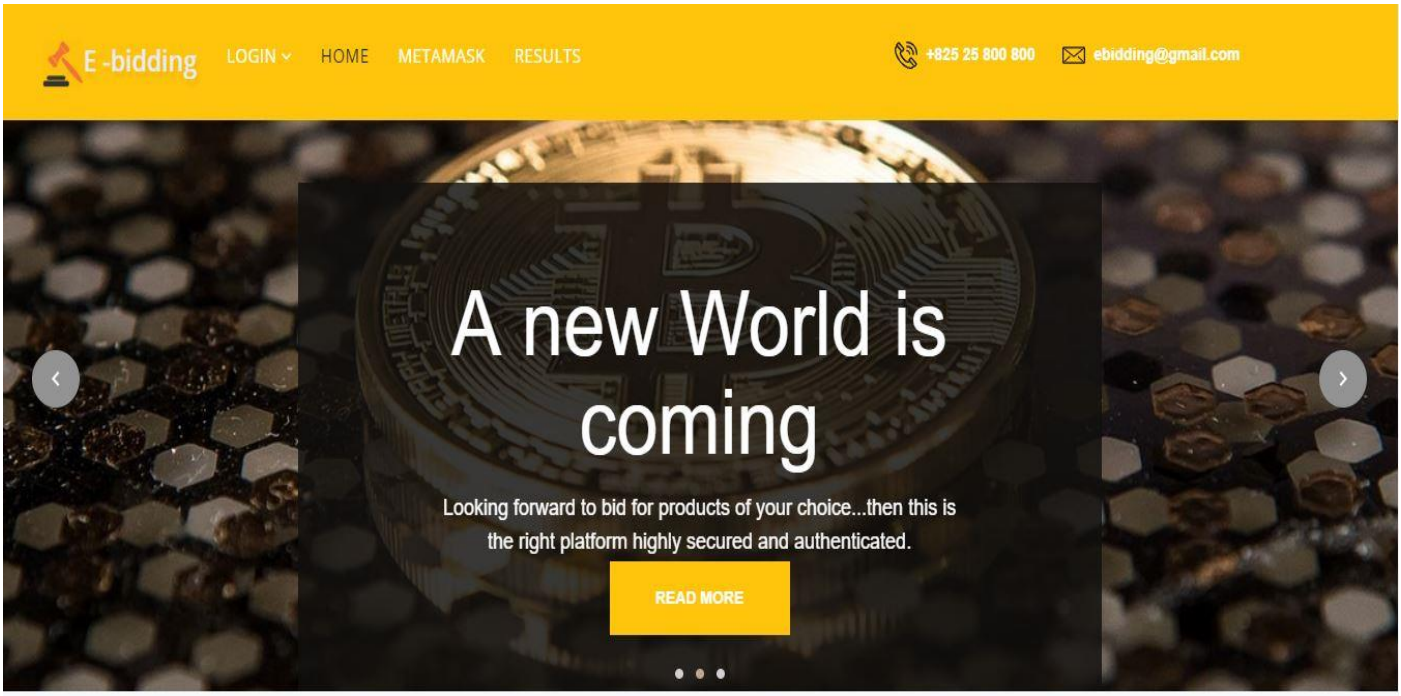
```
    function bid (uint productId,uint bidamt,string filename,uint BuyerId) public
    {
            require(productId > 0 && productId <= ProductCount);
            BidCount++;
            bids[BidCount] = Bid(productId,products[productId].sellerId,BidCount,bidamt,filename,BuyerId);
    }
    function addBidder(string _name,string _pass,string _salt,string _hash,address _add,string _passphrase,string privateKey) public
    {       BuyerCount++;
            bidder[BuyerCount]=Buyer(BuyerCount,_name,_add,_pass,_salt,_hash,_passphrase,privateKey);
    }
    function addFinalResult (uint resId,uint bide) public {
           require(resId > 0 && resId <= ResultCount);
           FinalResultCount ++;
           finalresults[ResultCount] = FinalResult(results[resId].sellerId,results[resId].prodname,results[resId].proddesc,results[resId].prodId,
            results[resId].prodTime,results[resId].resId,bide,FinalResultCount);
           results[resId].there="no";


    }
}
```

```
    uint public ProductCount;
    uint public BuyerCount;
    uint public ResultCount;
    uint public BidCount;
uint public FinalResultCount;
    constructor () public {

      }

    function addSeller (string _name,string pass,string passphrase,string _salt,string _hash) public {
           SellerCount ++;
           sellers[SellerCount] = Seller(SellerCount, _name,pass,passphrase,_salt,_hash);
    }
    function addProduct (uint sellid,string _name,string desc,uint amt,string time,string prodpath) public {
           ProductCount ++;
           products[ProductCount] = Product(sellid, _name,desc,amt,ProductCount,time,prodpath,"yes");
    }
    function updateProduct (uint prodId) public {
           require(prodId > 0 && prodId <= ProductCount);
           ResultCount ++;
           results[ResultCount] = Result(products[prodId].sellerId,products[prodId].prodname,products[prodId].proddesc,products[prodId].prodamt,
               products[prodId].prodId,products[prodId].prodTime,ResultCount,products[prodId].prodpath,"yes");
           products[prodId].there="no";


    }
```

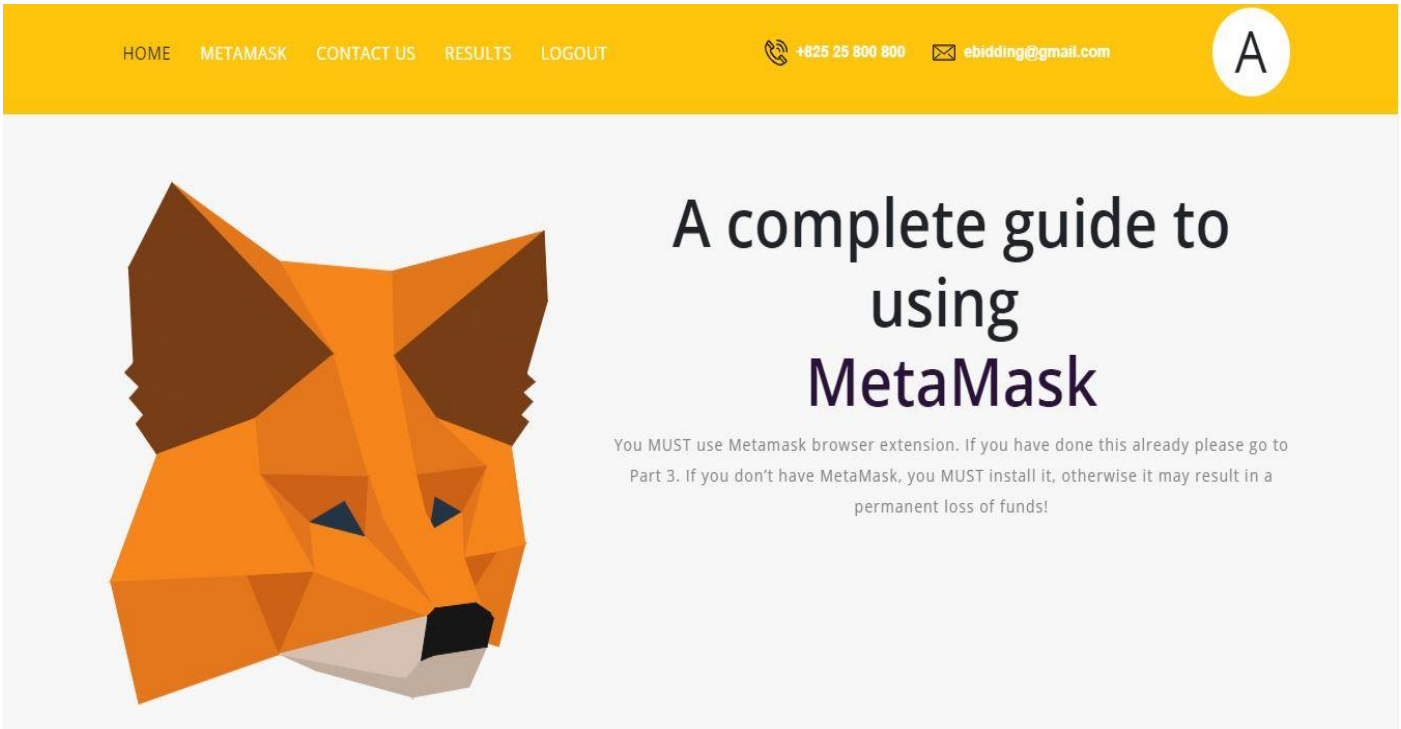
```
struct Result
{   uint sellerId;
    string prodname;
    string proddesc;
    uint prodamt;
    uint prodId;
    string prodTime;
    uint resId;
    string prodpath;
    string there;

}
struct FinalResult
{   uint sellerId;
    string prodname;
    string proddesc;
    uint prodId;
    string prodTime;
    uint resId;
    uint bidId;
    uint FinalResultId;

}

struct Bid
{    uint prodId;
    uint sellerId;
    uint bidId;
    uint prodamt;
    string filename;
    uint BuyerId;

}
 mapping(uint => FinalResult) public finalresults;
 mapping(uint => Product) public products;
 mapping(uint => Bid) public bids;
 mapping(uint => Seller) public sellers;
 mapping(uint => Buyer) public bidder;
 mapping(uint => Result) public results;
 // Store Candidates Count
 uint public SellerCount;
```

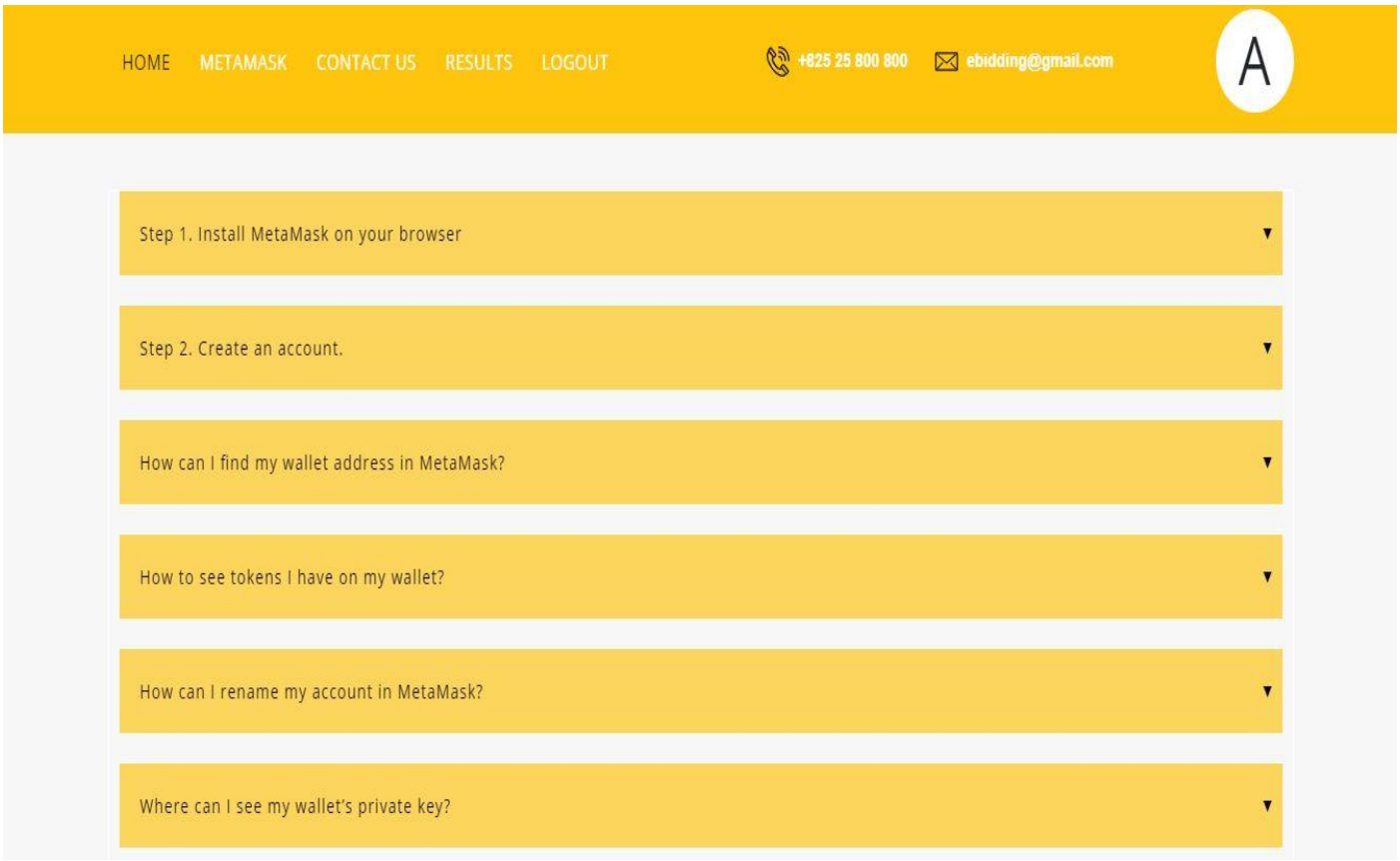
```
pragma solidity ^0.4.2;

contract Auction {
    // Model a Candidate
    struct Seller
    {
        uint id ;
        string name;
        string pass;
        string passphrase;
        string salt;
        string hash;
    }
    struct Product
    {
        uint sellerId;
        string prodname;
        string proddesc;
        uint prodamt;
        uint prodId;
        string prodTime;
        string prodpath;
        string there;
    }
    struct Buyer
    {   uint id;
        string name;
        address add;
        string pass;
        string salt;
        string hash;
        string passphrase;
        string privateKey;
    }
    struct Result
    {   uint sellerId;
        string prodname;
        string proddesc;
        uint prodamt;
        uint prodId;
        string prodTime;
```

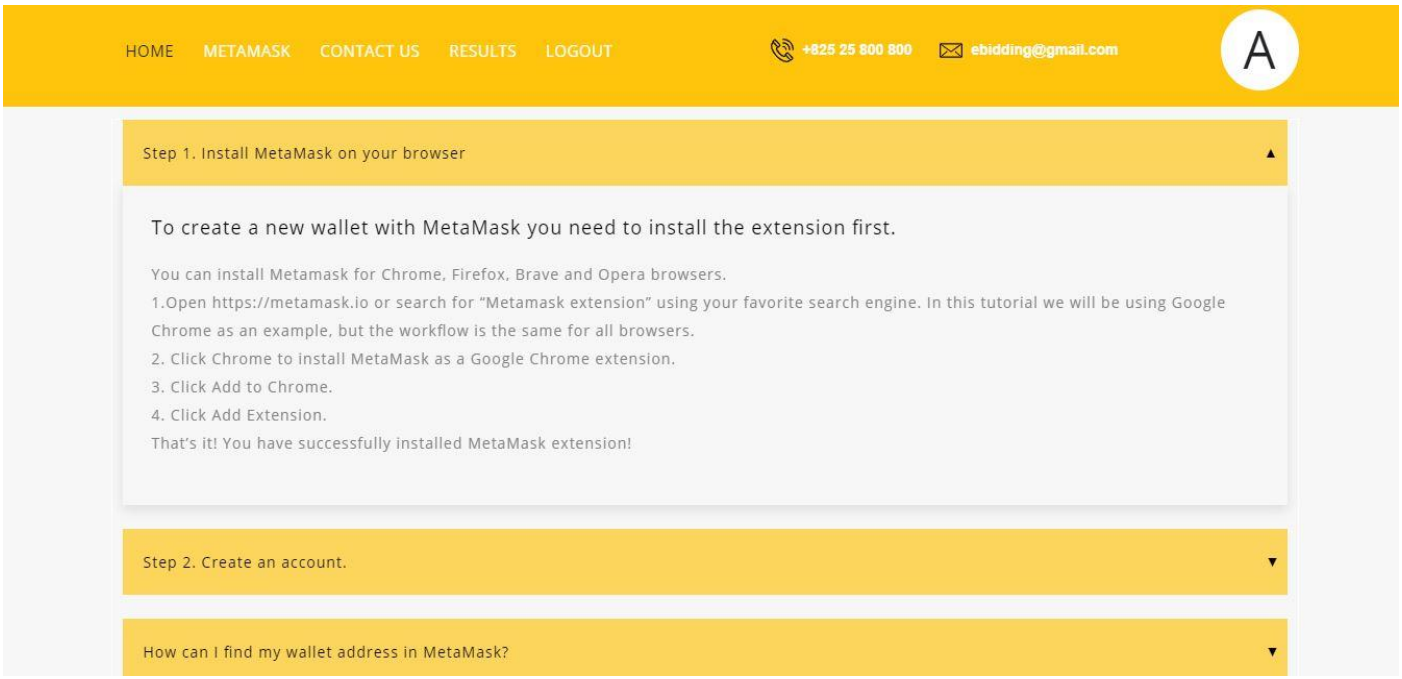
HOME PAGE



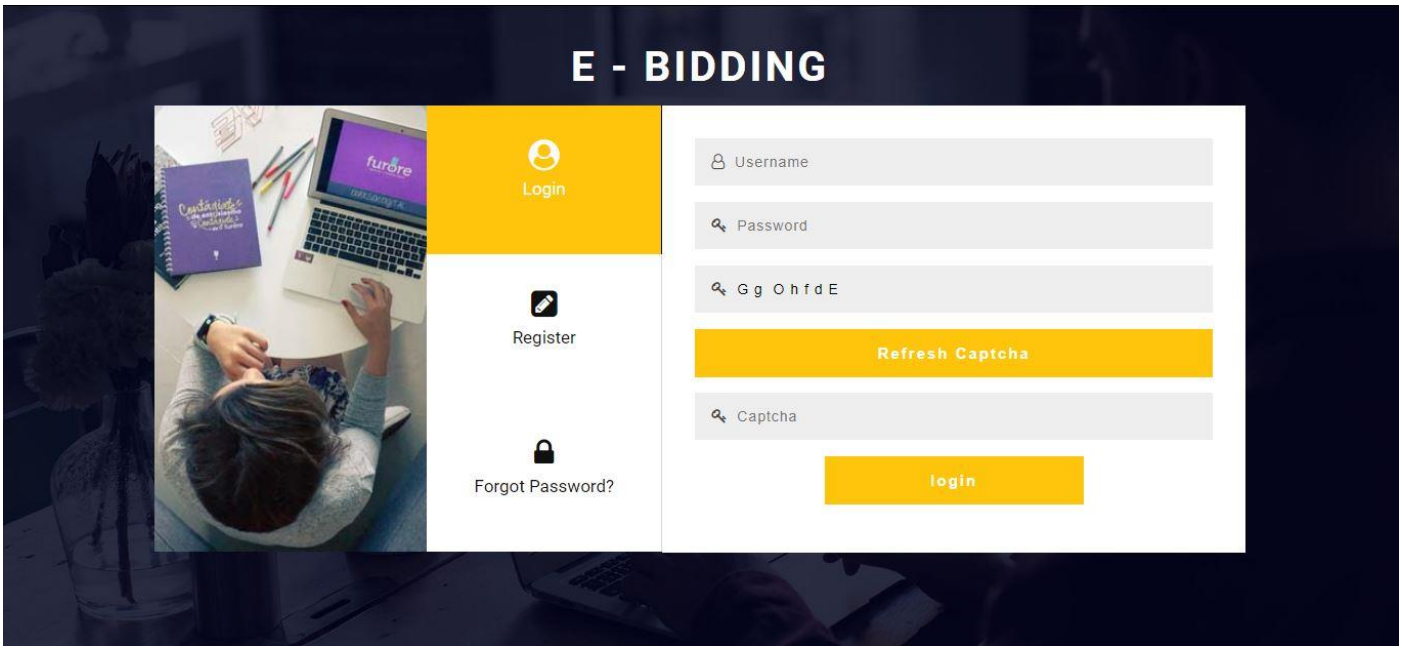
METAMASK INFORMATION



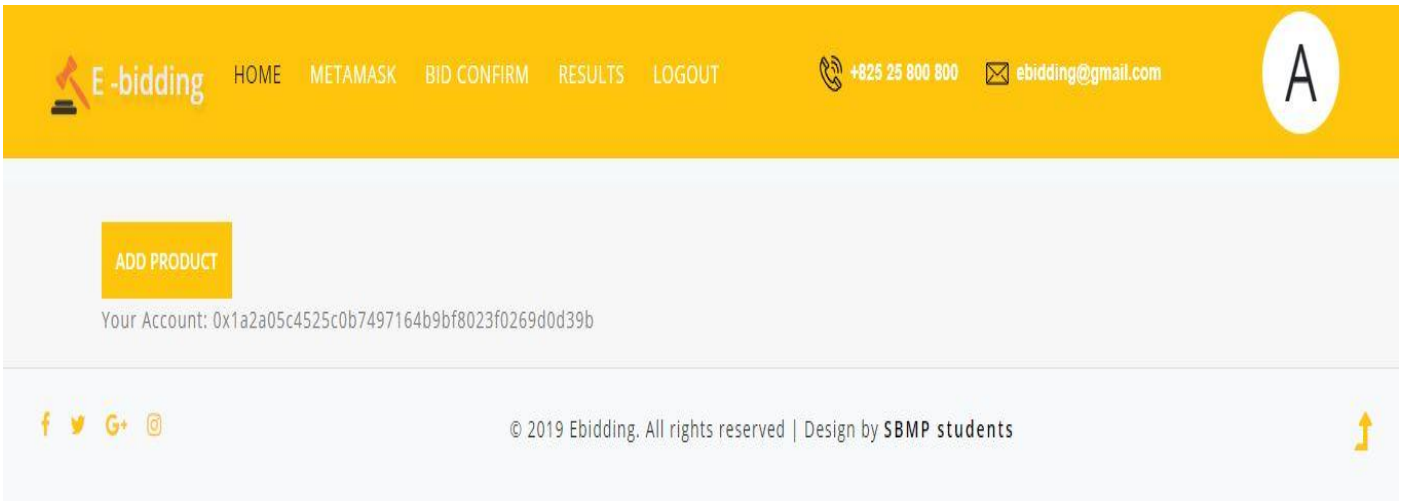

HOME
METAMASK
CONTACT US
RESULTS
LOGOUT
+825 25 800 800
ebidding@gmail.com
A
Step 1. Install MetaMask on your browser
Step 2. Create an account.
How can I find my wallet address in MetaMask?
How to see tokens I have on my wallet?
How can I rename my account in MetaMask?
Where can I see my wallet's private key?

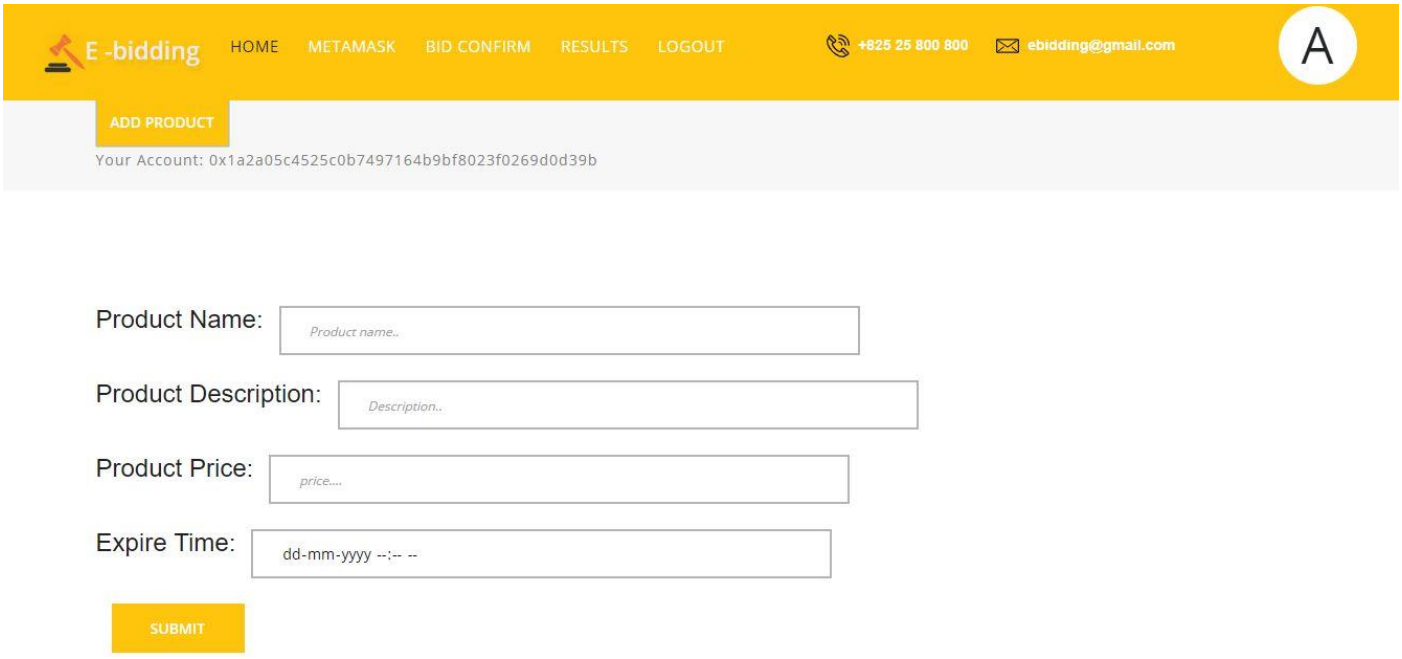


REGISTRATION

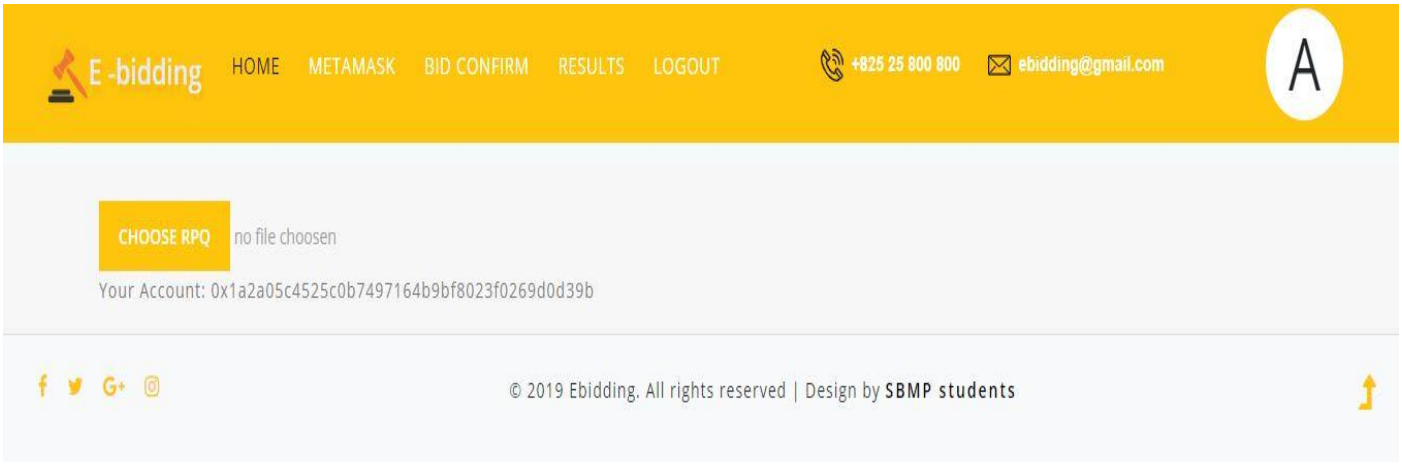


ADD TENDER

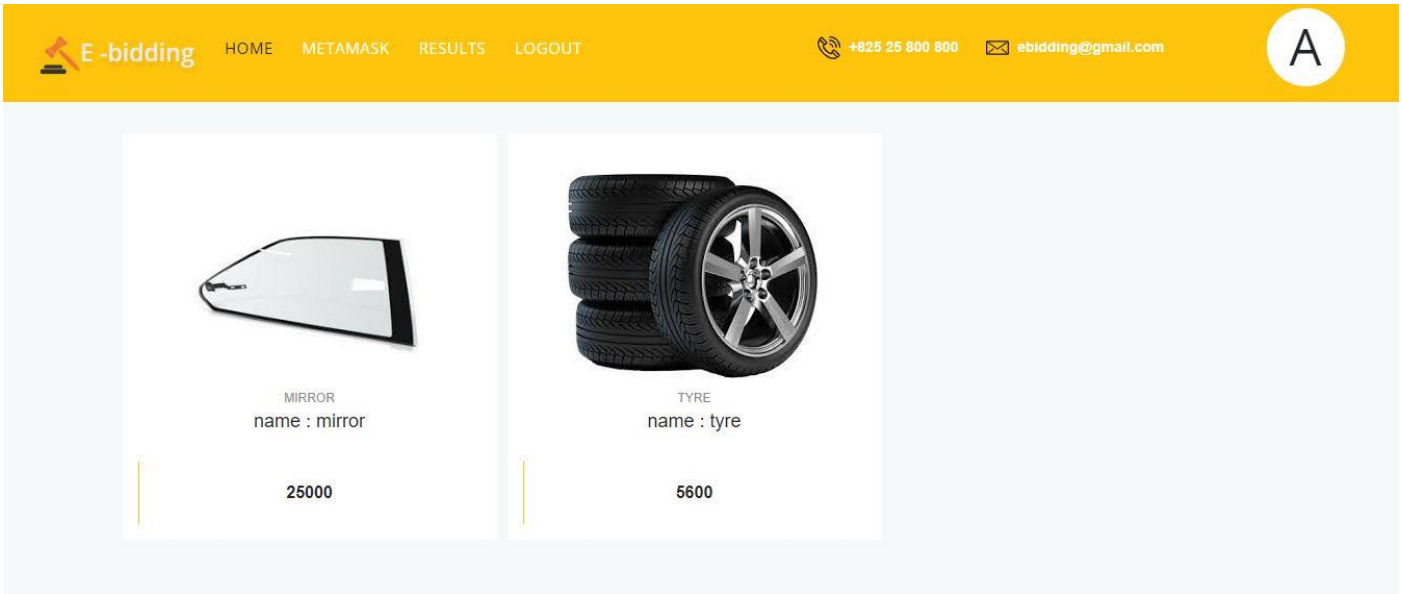

E -bidding
HOME
METAMASK
BID CONFIRM
RESULTS
LOGOUT
+825 25 800 800
ebidding@gmail.com
A
ADD PRODUCT
Your Account: 0x1a2a05c4525c0b7497164b9bf8023f0269d0d39b
© 2019 Ebidding. All rights reserved | Design by SBMP students

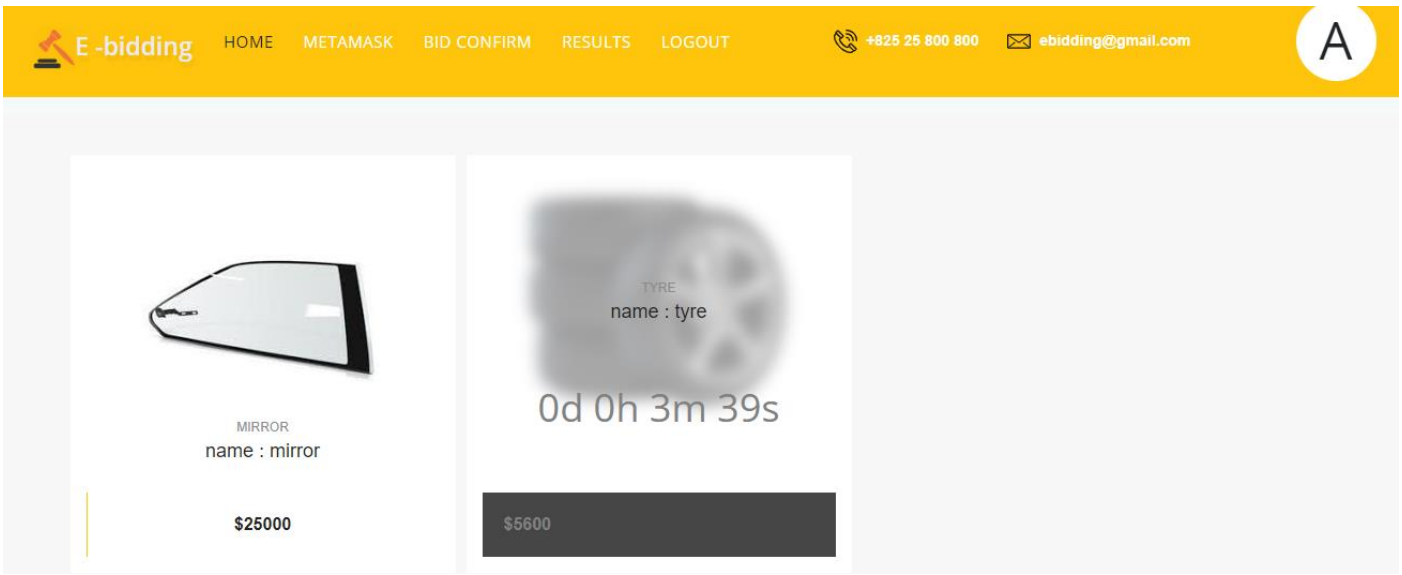

E -bidding
HOME
METAMASK
BID CONFIRM
RESULTS
LOGOUT
+825 25 800 800
ebidding@gmail.com
A
ADD PRODUCT
Your Account: 0x1a2a05c4525c0b7497164b9bf8023f0269d0d39b
Product Name:
Product name..
Product Description:
Description..
Product Price:
price....
Expire Time:
dd-mm-yyyy --:-- --
SUBMIT

CHOOSE RFQ

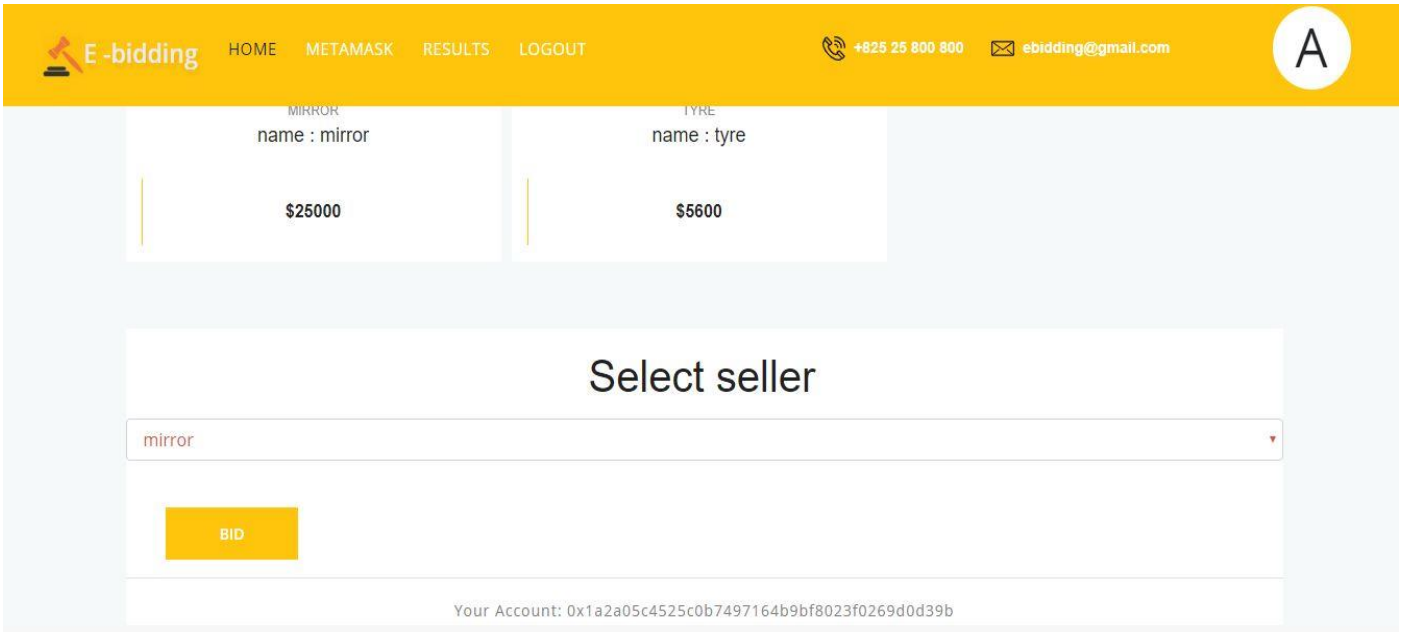


SELLER DISPLAY

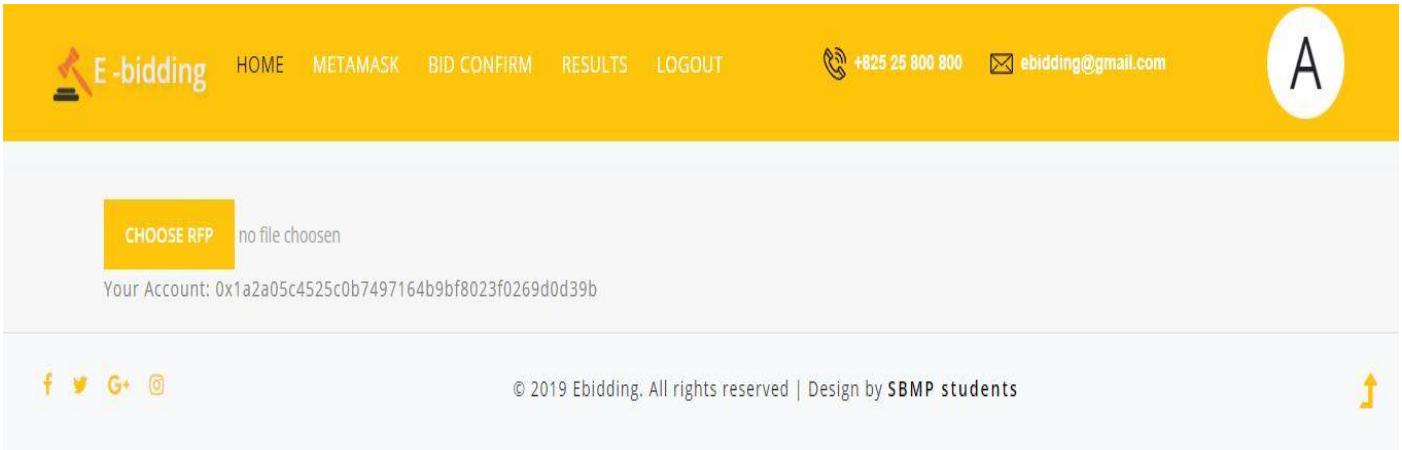




ADD QUOTATION



CHOOSE RFP


E -bidding
HOME
METAMASK
BID CONFIRM
RESULTS
LOGOUT
+825 25 800 800
ebidding@gmail.com
A
CHOOSE RFP
no file choosen
Your Account: 0x1a2a05c4525c0b7497164b9bf8023f0269d0d39b
© 2019 Ebidding. All rights reserved | Design by SBMP students

TESTING

15.Testing

We have conducted many different types of testing throughout are project:

- Unit Testing

  At the end of every week we have conducted a unit testing on separate modules of the project, discovered bugs, raised issues on those bugs and resolved the bugs over the weekend.

- Integration Testing

  After developing any major part of a program we conducted an integrations testing going through the entire workflow, such as registering users and then logging in, uploading data, images and then checking over the functionalities.

- System Testing

  After developing the whole system we performed system testing wherein we conducted testing the working of code, internal working tried various possibilities by providing different input and output.

- Acceptance Testing

  At the end we performed acceptance testing wherein we tried it running on different computer devices within a network by providing localhost, port number while remotely accessing the application and solved the problems in accessing it.

ADVANTAGES

10. Advantages

- Easy to use.
  This application can be accessed from anywhere, anytime around the world.

- Integrity is maintained.
  This application hashes data and performs proof-of-position to prevent data from being hacked.

- Processing is faster.
  This application is decentralized and can be remotely accessed as Blockchain technology is used so processing of operations is faster.

- No third party involvement
  Blockchain technology removes the third-party verification by making it decentralized i.e. accessible within a network.

- Security is increased.
  This application provides automated third-party authentication by the system itself and keep data in hashed format so that data cannot be hacked easily.

LIMITATIONS

11. Limitations

- Metamask must be there on client's computer for providing digital wallet to ethers provided by ganache and use them for performing transaction/operations in this web application.
- Continuous internet connection is required as continuous updates are to be recorded on server side such as adding RFQ, RFP, bidding, viewing the results etc.
- This application is particularly designed for Car Companies requiring spare car parts in order to provide efficiency to users.
- This application is only available for computer devices because functionalities are added according to the requirement of organization selling car products which they can access through computer devices as organizations commonly use them.
- Users must be aware of accessing the application as different framework and languages are used.
- To use the application remotely IP address/localhost must be provided in the URL which must be added explicitly in URL to initiate the auction.
- This application can be accessed by people within a network .i.e. within organization, companies etc.

APPLICATION

12. Applications

- It can be used for conducting auctions of car products.
- Used for communication between two Companies for selling and purchasing car products.
- It can be used for making tender process automated and secure.
- It is used where time saving is important.
- It is used where no third party involvement is required.
- It can be used for saving space as it is distributed due to use of Blockchain technology.
- Used for providing security to the process of adding of RFQ, RFP, PO.
- It can be used for Car companies requiring faster tender selection, rejection, processing etc.

FUTURE SCOPE

13. Future Scope

- It can be used for any kind of auction such as clothes, properties or products specific to organization requiring tender to release through computer devices remotely.

- More security can be added by using RSA as encryption algorithm because in Blockchain technology hashing and digital signature is implicitly used to maintain integrity.

- It can be used for centralized systems also.

- It can be made accessible through mobile devices by adding further functionalities.

- The Purchase Order verification and acceptance process can be made automatic.

- This application can be made accessible through other operating systems by adding modules and functionalities required for them.

- This application can also be implemented with React js and databases such as Firebase, Mongo dB can also be integrated to store large-scale of data and contracts can be deployed on other networks also depending on the needs of the user.

- This application can be made accessible to other outside networks by changing settings within ganache and MetaMask.

CONCLUSION

14. Conclusion

- We took this project to improve auction process and provide better access to the Car Company by providing them flexibility to add tenders according to their requirement by removing the need for manual process of conducting auction and eliminated the need for manual tender adding process saving time and money.

- Along with the application's general details described in this report, a major aspect of this project is the code documentation, security provided and understanding of sellers, bidders, and other Companies who are going to use, maintain and enhance this project in future.

- Following the theory of making auction process automated and easier for Car Company we created this application to increase efficiency and security by using Blockchain as a technology.

REFERENCES

15. References

1. https://www.ibm.com/developerworks/community/blogs/9b7ea9e9-c769-4552-aae7-f3ce7f463c5b/resource/BLOGS_UPLOADED_IMAGES/blockchain-technology-decentralized-disruption-4-638.jpg

2. https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2018/05/Advantages-of-Blockchain-01.jpg

3. https://ieeexplore.ieee.org/document/7918009

4. https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

5. https://medium.com/programmers-blockchain/importing-gson-into-eclipse-ec8cf678ad52

6. https://metamask.io

7. https://www.tutorialsteacher.com/nodejs/nodejs-modules

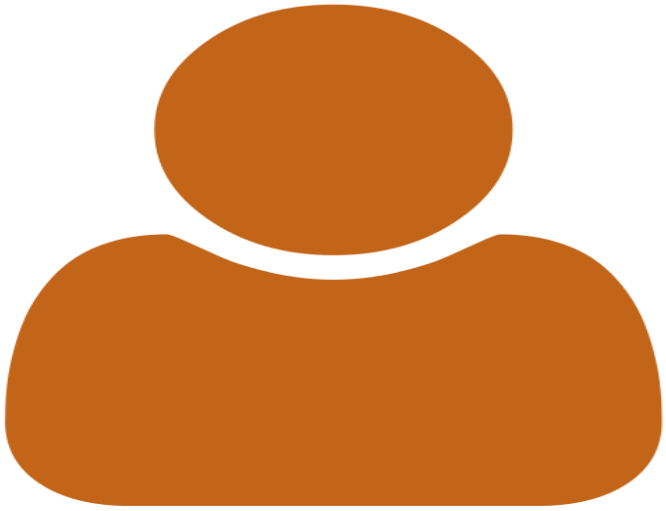
8. https://www.npmjs.com

9. https://github.com/trufflesuite/truffle

10. https://www.researchgate.net/publication/318131748_An_Overview_of_Blockchain_Technology_Architecture_Consensus_and_Future_Trends

11. https://www.coindesk.com/information/what-is-blockchain-technology

12. https://arxiv.org/ftp/arxiv/papers/1801/1801.03528.pdf



ABOUT US

16. About us

This project has been developed by group members of *"Group 2",*
Who believe that hard work is the key to success and interested in learning new technologies with the goal to provide best possible technological solutions to reduce problems in society.

1698002 - Shweta H. Chudasama

1698011 - Vidhi D. Jain

1698037 - Ankita V. Tripathi