DataRitz Technologies
Enhancing Technology Experience

OWNERSHIP CHAIN

Summer Internship Project Report

Submitted

To

Data Ritz Technologies

Duration : 6 weeks

By

KHUSHI GARG
1803210081

Under the guidance of
Dr. Bhubaneshwar Prasad Sharma

DataRitz Technologies
Enhancing Technology Experience

DataRitz Technologies
Enhancing Technology Experience

CERTIFICATE

This is to certify that Project Report entitled "**Ownership Chain**" which is submitted by in partial fulfillment of the team **The Quad Squad** requirement for the summer internship of degree B. Tech. in Department of **Computer Science and Engineering** of **ABES Engineering College** is a record of the candidate own work carried out by him under my/our supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

Supervisor : Dr. Bhubaneshwar Prasad Sharma

Date : 03/07/20

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech summer internship Project undertaken during B. Tech. Second Year. We owe special debt of gratitude to Professor Name ***Dr. Bhubaneshwar Prasad Sharma****, DataRitz Technologies for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.*

We also take the opportunity to acknowledge the contribution of Head, Department of ***DataRitz Technologies*** *for his full support and assistance during the development of the project.*

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the DataRitz Technologies for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature:

Name : Khushi Garg

Roll No : 1803210081

Date : 03/07/20

OWNERSHIP CHAIN

TABLE OF CONTENTS

1.1ProjectDescription

- Whenever we buy anything, a bill is given to us which is the proof that particular thing belongs to us.
- But many times bills are lost or the content of the bill fades away so we have no proof of our belongings.
- Now the need is to have a system in which all our bills are stored permanently for future also.
- Ownership Chain is a “new” approach for bill management for storing bill in the database that is immutable.
- The project is basically based on the immutability power of Blockchain.
- The whole system is user sided and is maintained by the user.
- User can add the image and details of bills and products.
- Those details then will get stored on blockchain permanently and user can access them whenever he wants.

1.2 Project Need

- The things we purchased in our daily lives includes bill as a proof.
- But sometimes bills can get lost or the content written on the bill fades.
- So this project helps in protecting the ownership of our products by saving our bills.

1.3 User Story

- User has to first sign up and has to login to access the account.
- User can add their bills and view all the previously added bills.
- User can add the image of bill and product for authentication.
- User can view and download image of previously added bills.
- The role of system is to confirm user via email that whether the bill is added or not.

1.4 Team

The name of team is **The Quad Squad.** Team members are:

1. Priyanshi Garg (lead)
2. Sakshi Singh
3. Khushi Garg
4. Simran Sirohi

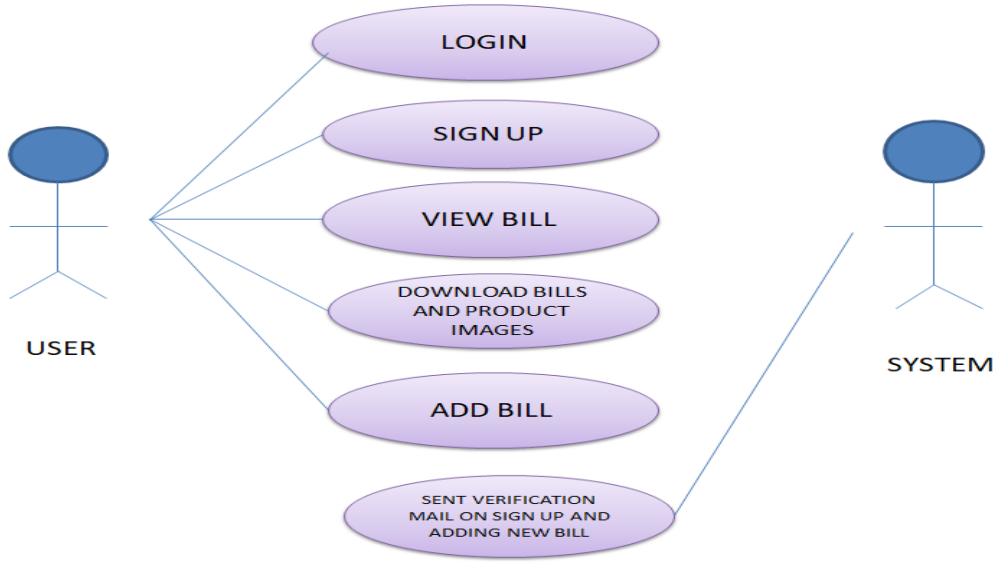
1.5 Entities

There are only two entities involved in this project:

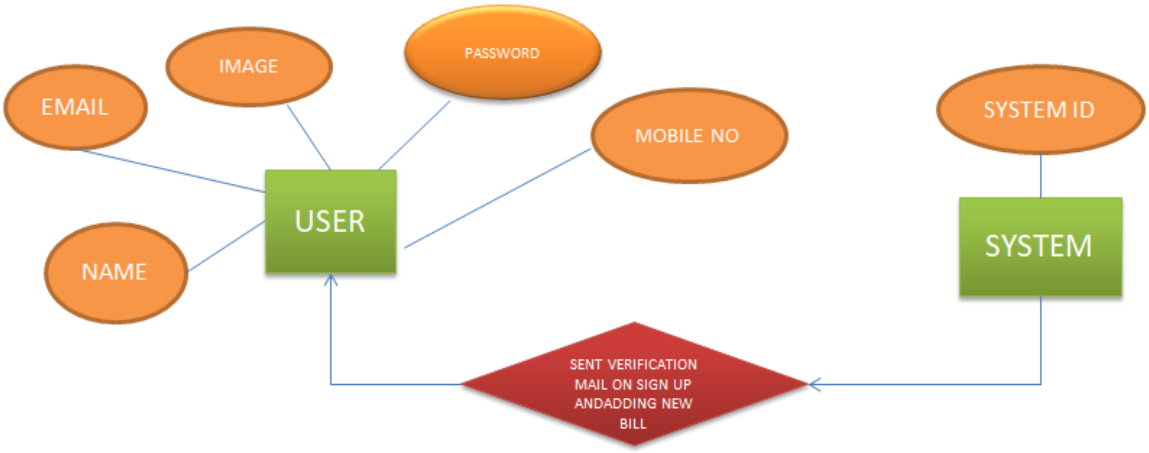
- ***User :*** The user has to first signup and sign in to create the account. The user can add and view all the previously added bills.

- ***System :*** The system has to authenticate the user via sign in and has to send an email whenever a new bill is added in the system.

1.6 Use Case Diagram

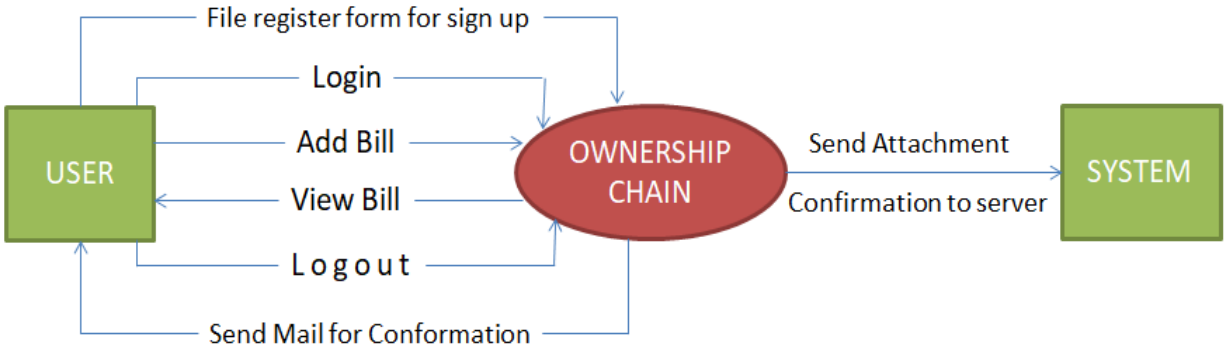


1.7 ERDiagram

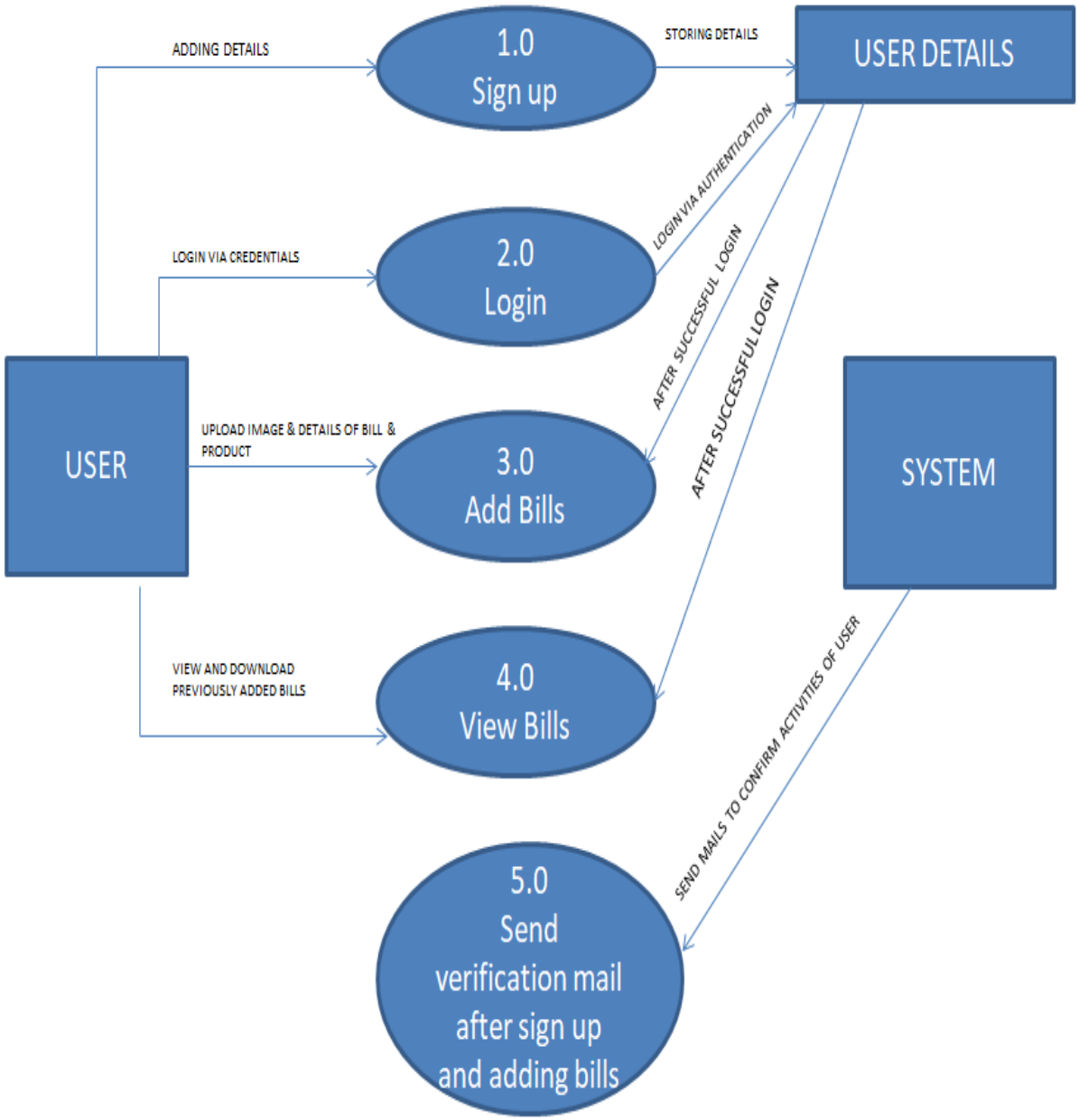


1.8 Data Flow Diagram

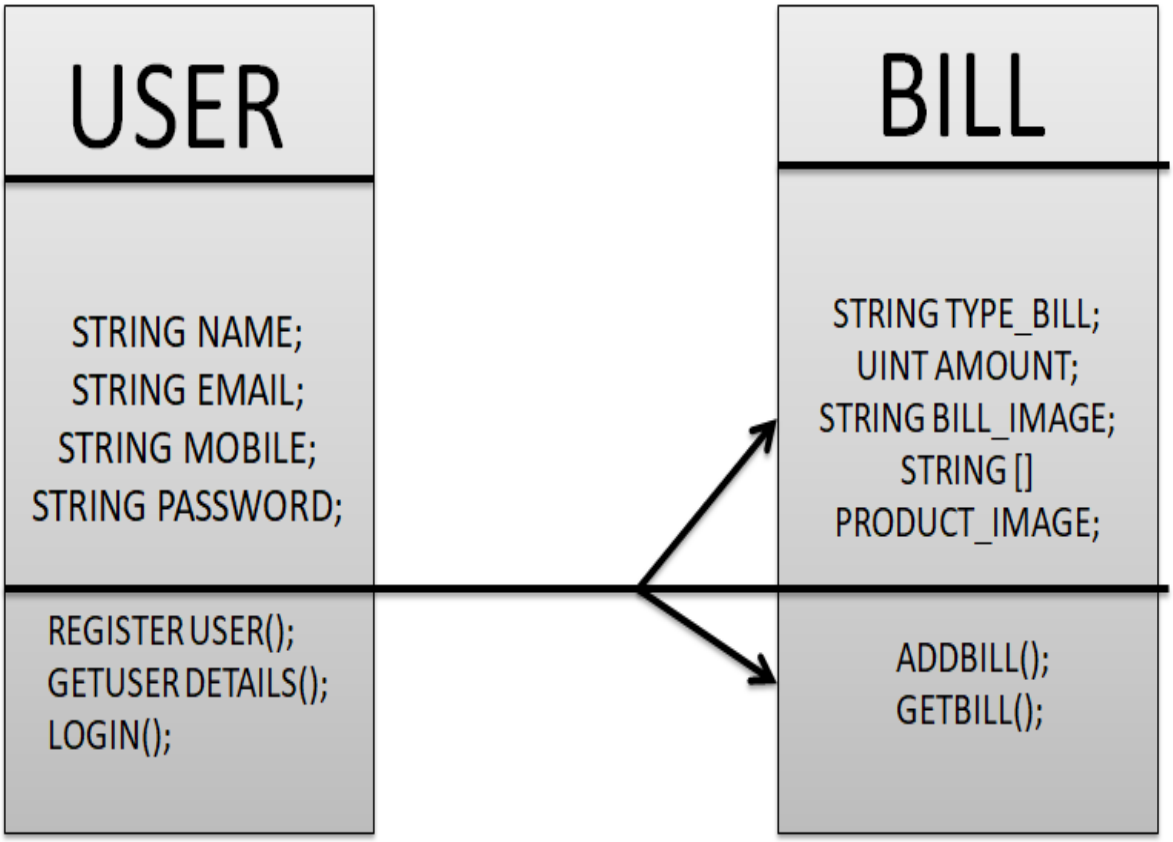
1. 0^{th} level dfd



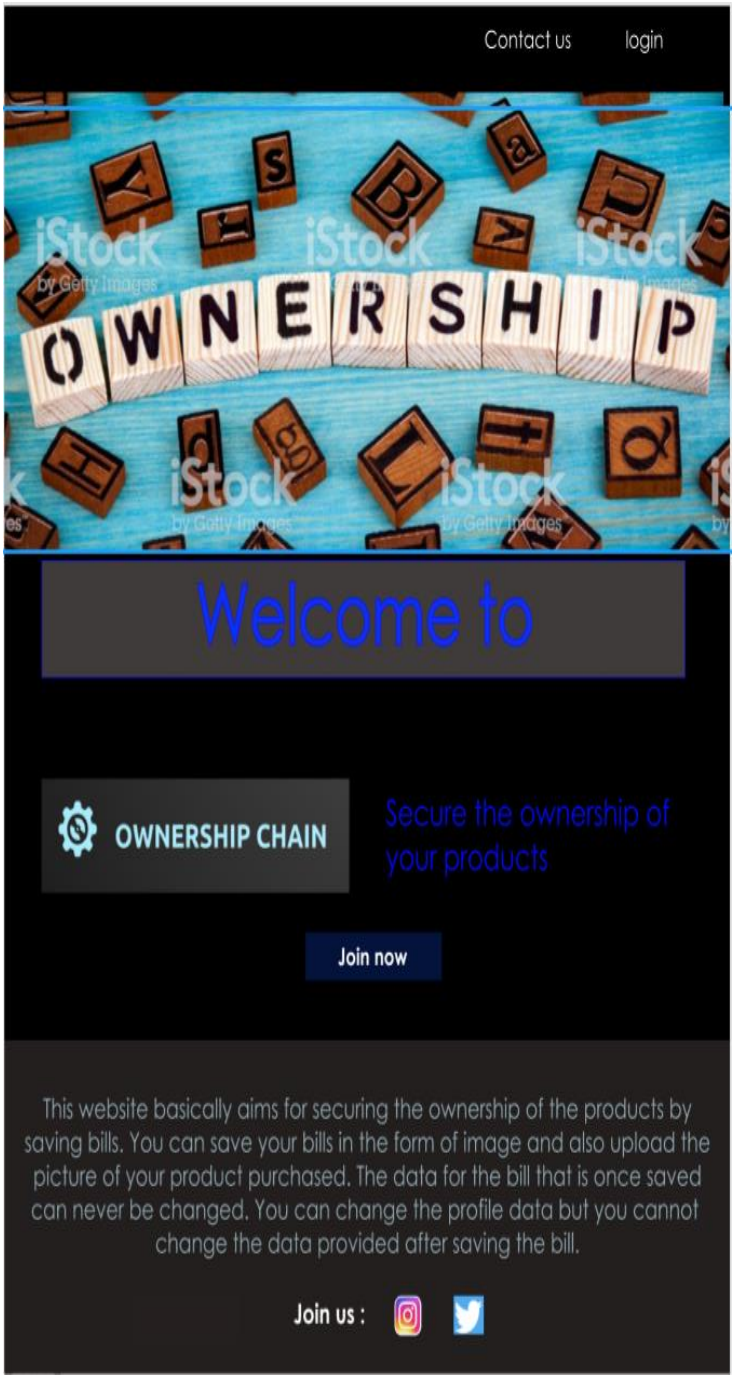
1st level dfd



1.9 Class Diagram



1.10 Prototypes

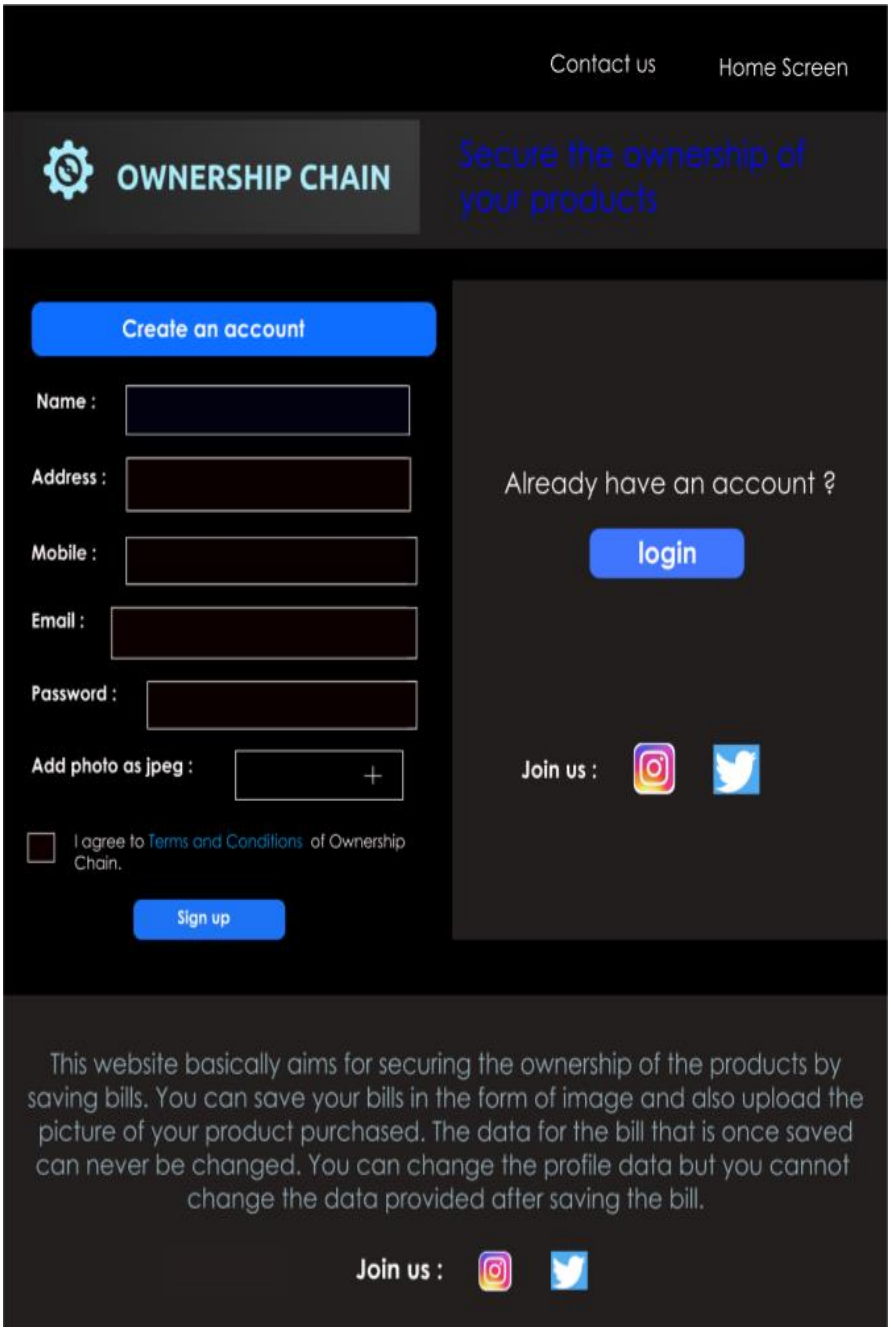


Controls

- It is the welcome screen of our website.
- It contains only one button i.e.,
- Join now.
- One has to click on join now to access website further.
- There are two links for Contact us and login.
- Login will direct to login page and Contact us will open gmail with receiver as official mail id.

Rules

- There are no rules for this page.

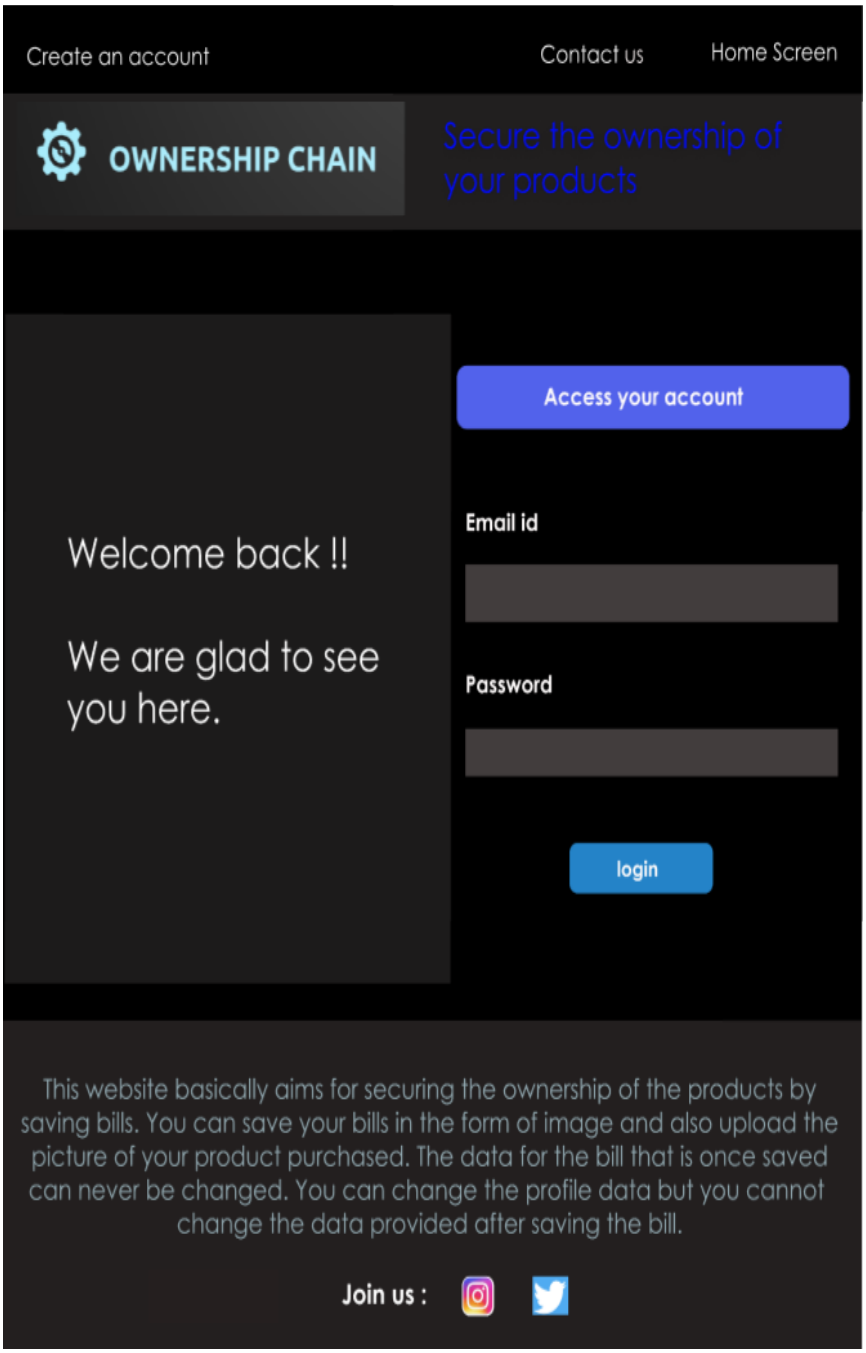


Controls

- This page is used for the users who are new to the site.
- This page stores information and login credentials for a user.
- This page also contains login button.

Rules

- Name box should contain alphabets only.
- Address can contains alphabet, numerals, comma, slash and hyphen.
- Mobile contains only 10 digits.
- Email contains numerals, alphabet, dot and @.
- Password contains alphabets, numerals

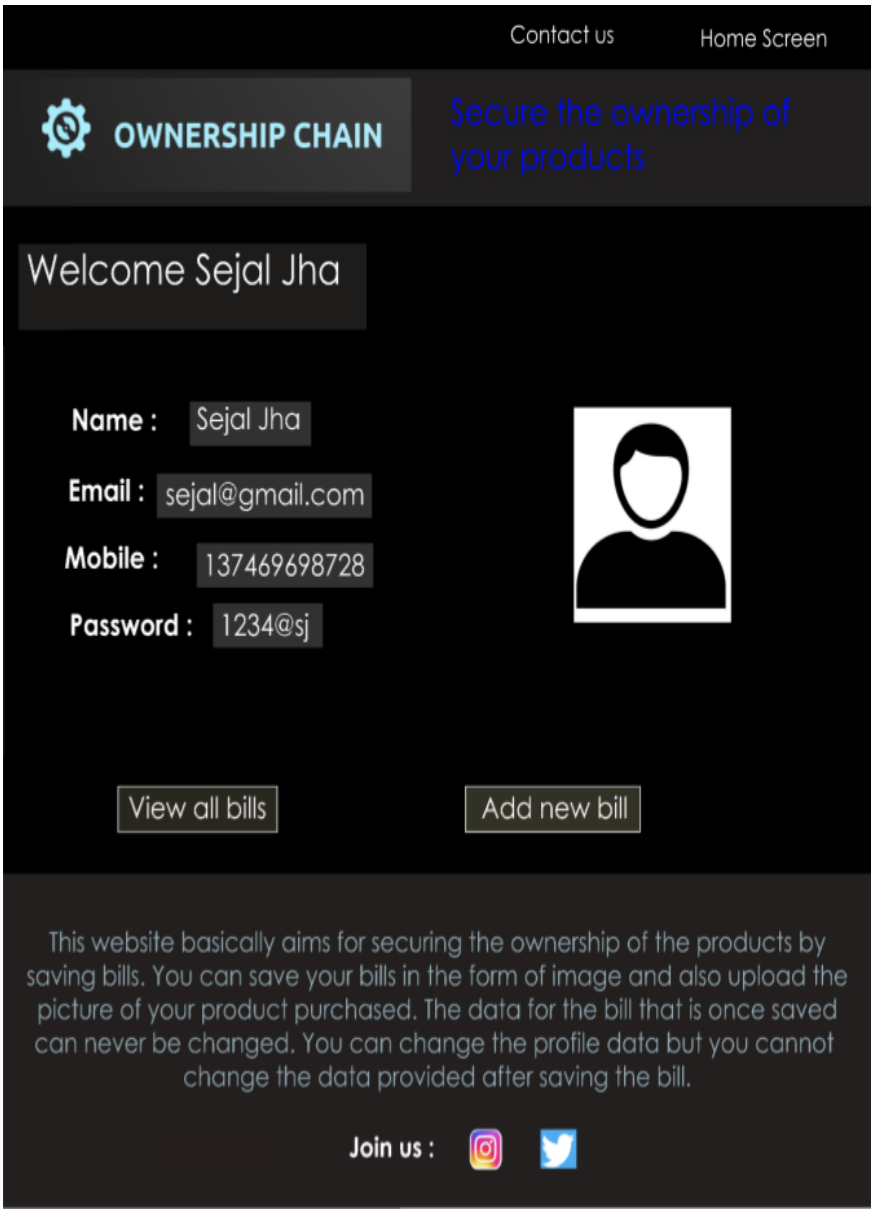


Controls

- This screen is the login screen and user needs to enter credentials to access the account.

Rules

- Email id and password need to be entered as entered on the time of sign up.

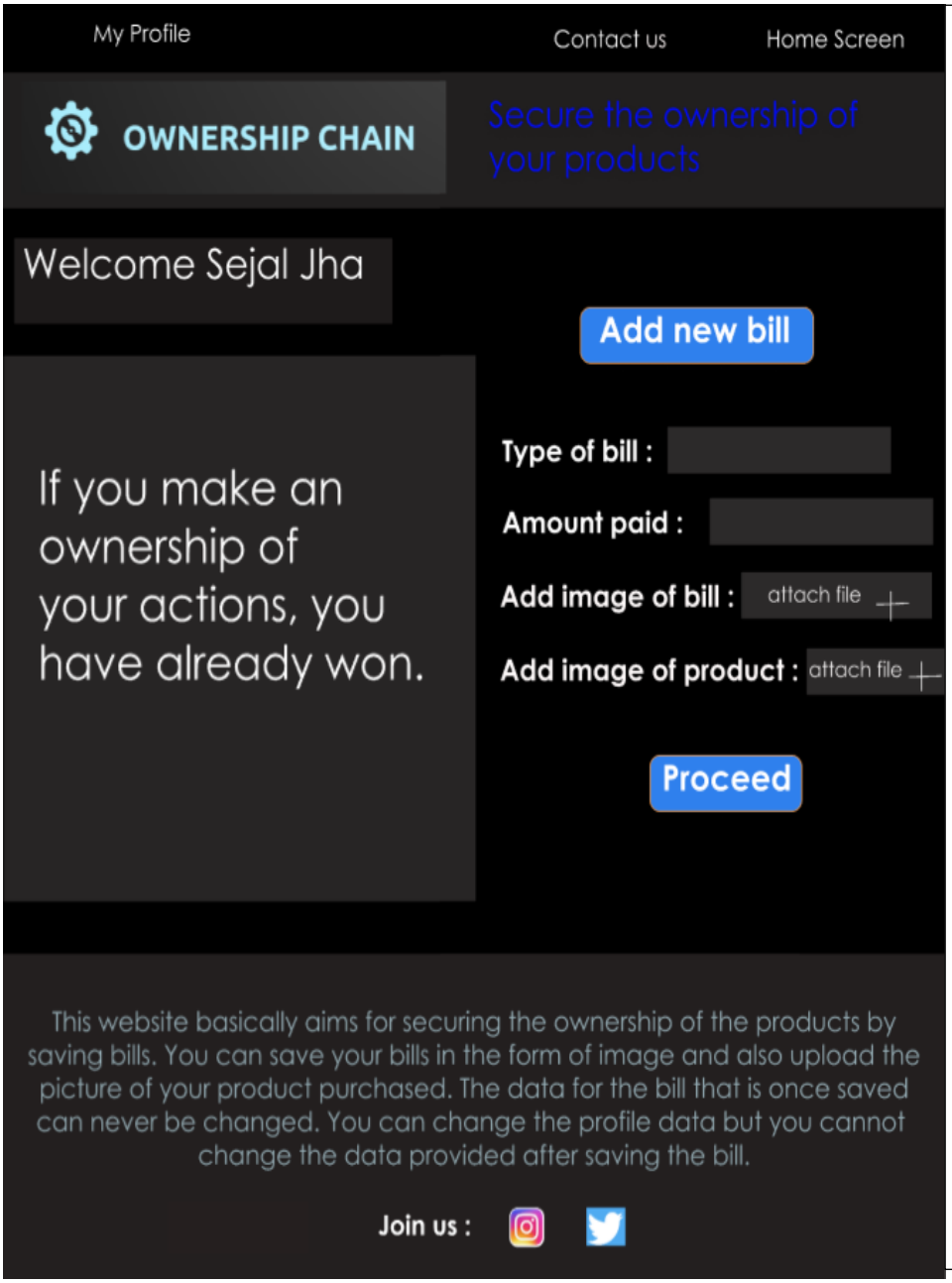


Controls

- This page has two controls i.e, View all bills and Add new bill.

Rules

- There are no defined rules for this page.

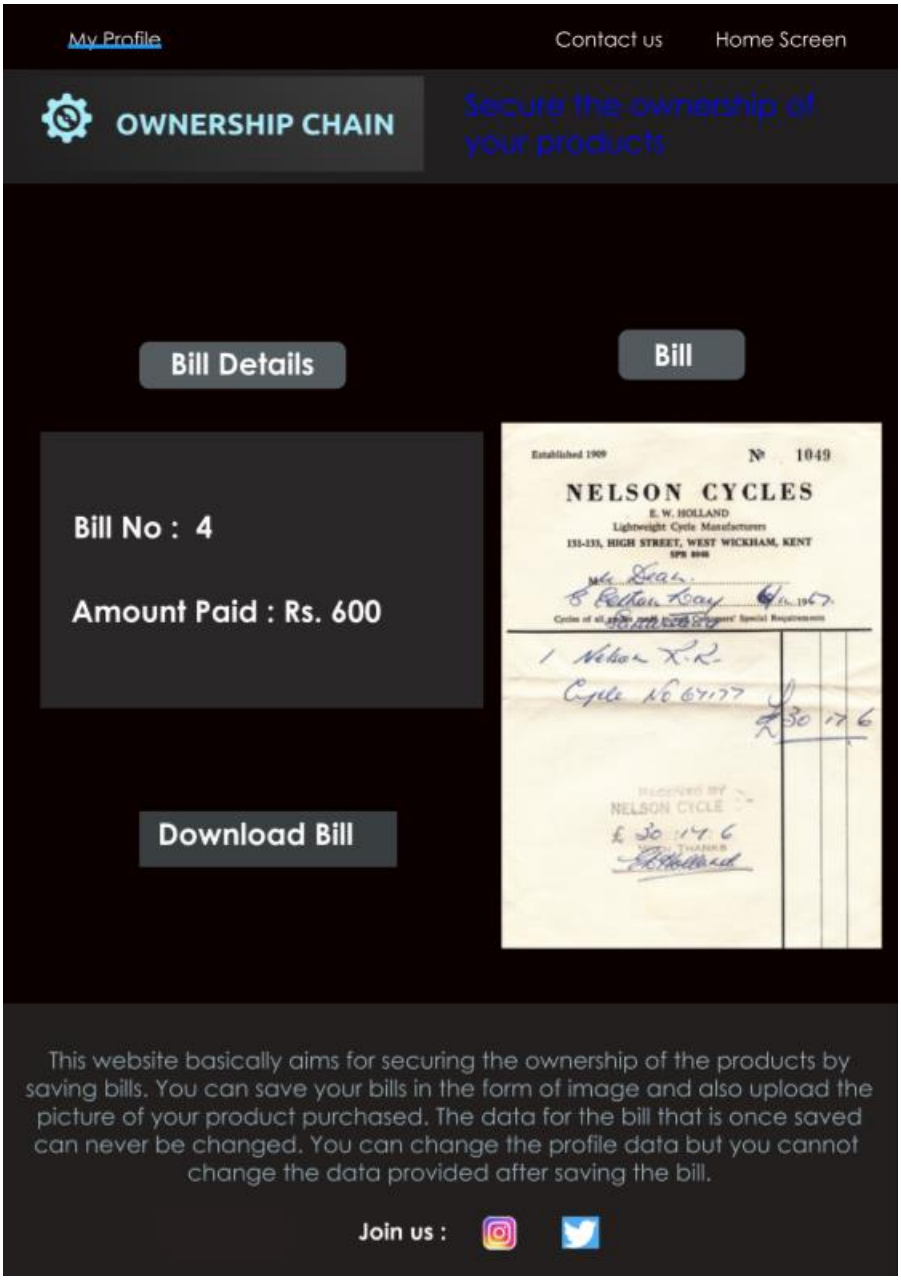


Controls

- This is the add bill page of the user.
- This page contains information about the newly added bill.

Rules

- The type of bill contains alphabet.
- Amount paid contains alphabets, dot and numerals.
- On clicking proceed, the info entered cannot be

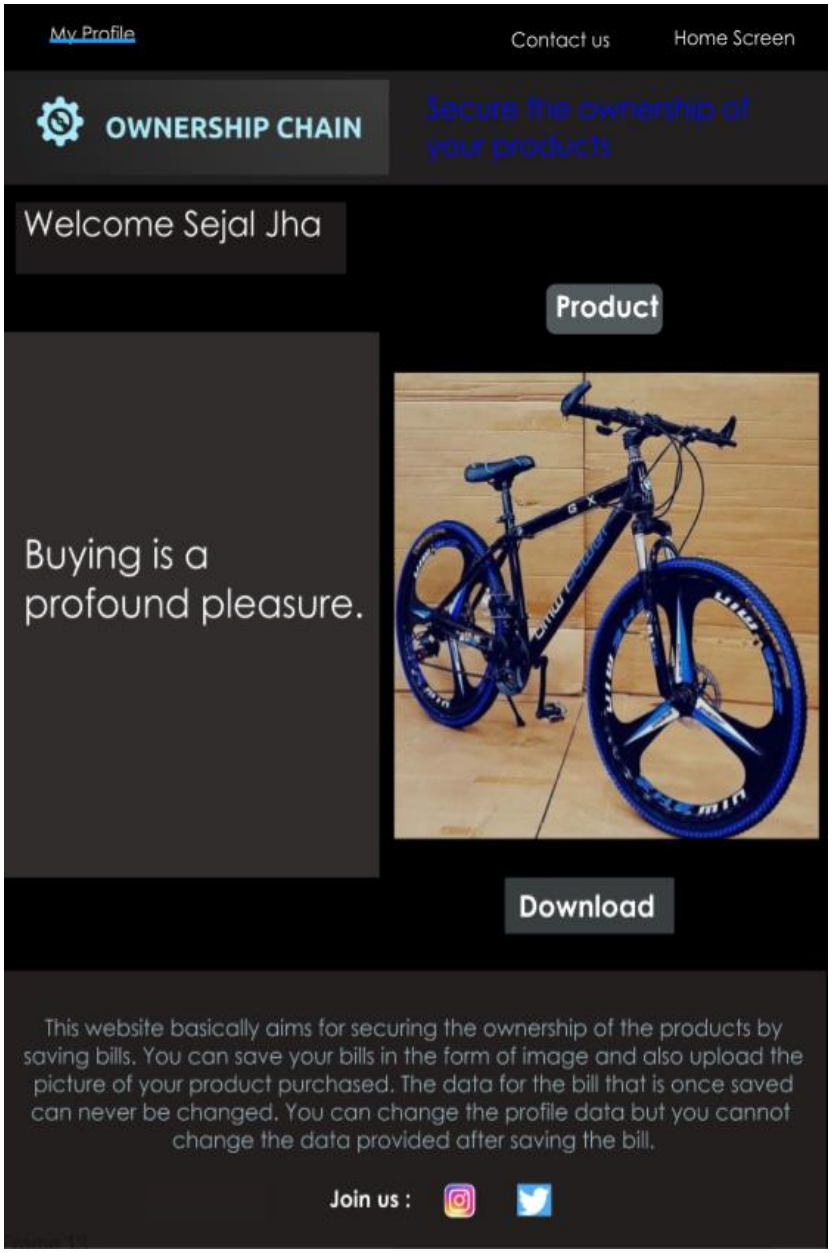


Controls

- User can download the bill from here.

Rules

- There are no defined rules for this page.

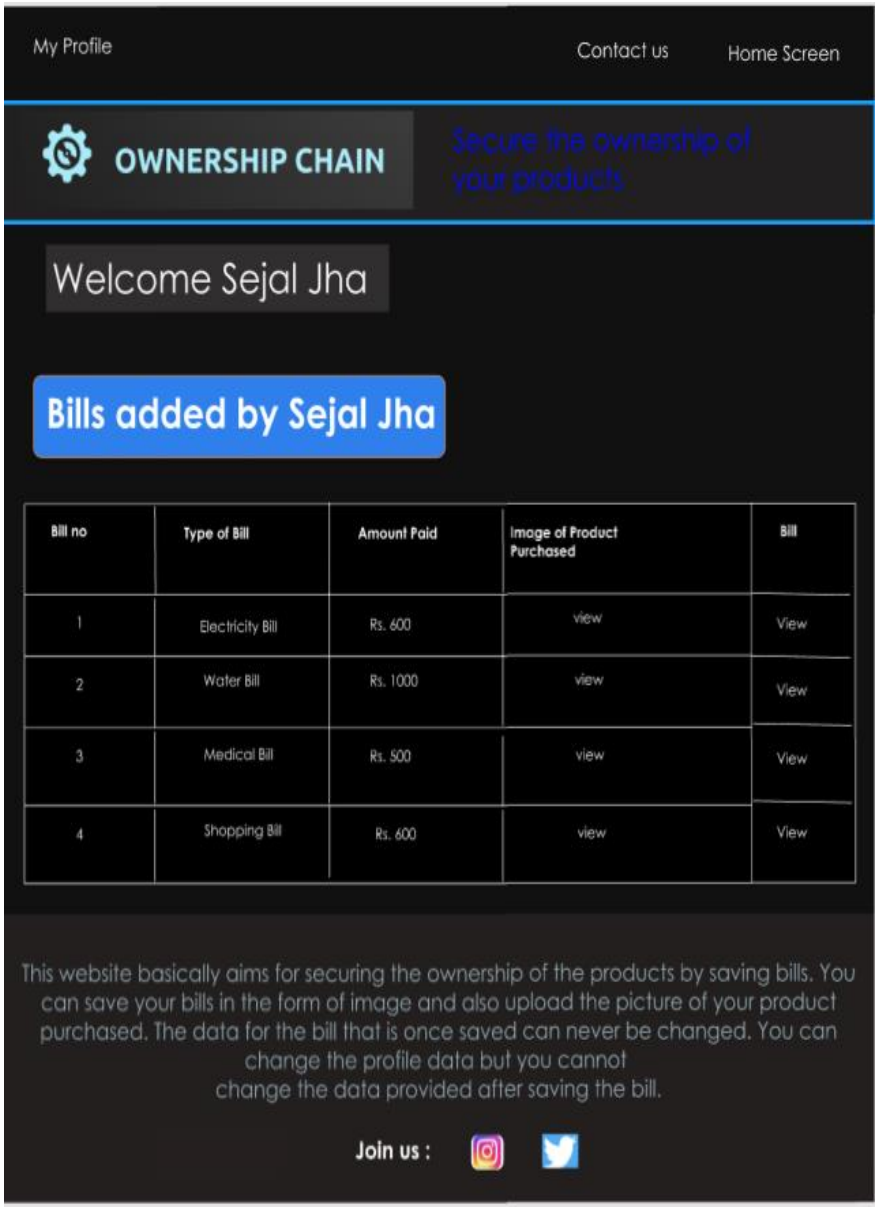


Controls

- This is the image of the products purchased.
- User can download the image .

Rules

- There are no defined rules for this page.



Controls

- This is the page where one can view all their bills they have added till now.
- User cannot edit the data.

Rules

- There are not any defined rules for this page.
- On clicking view, user can view and download the image.

1.11 Smart Contract Code

```
pragma solidity ^0.4.26;

contract TestContract
{
    struct User
{
        string name;
        string email;
        string mobile;
        bytes32 password;
    }

    struct bill
{
        string type_bill;
        uint amount;
        string bill_image;
        string product_image;
    }

    address owner;

    uint i=0;

    mapping(string=>bytes32) e_p;
```

```
mapping(address=>bill[]) a_b;

 User []u;

 function registerUser(string _name,string _email,
string _mobile, string _password) public
{

User memory user=User(_name,_email,_mobile,
keccak256(abi.encodePacked(_password)));
//keccak256, sha3, sha256
        u.push(user);
        e_p[user.email]=user.password;

    }

 function getUserDetails() public view
returns(string,string,string,bytes32)
{

return(u[0].name,u[0].email,u[0].mobile,u[0].pass
word);
    }

    function login(string _email,string
_password)constant public returns(bool)
{
```

```
if(e_p[_email]==keccak256(abi.encodePacked(_password)))
            return true;
        else
            return false;
    }

    function addBill(string _type, uint _amount, string _bimage, string pimage) public
{
        bill memory Bill = bill(_type,_amount,_bimage,pimage);
        a_b[owner].push(Bill);
        i++;
    }

    function getbill() public view returns(string, uint, string, string)
{
        return (a_b[owner][i-1].type_bill,a_b[owner][i-1].amount,a_b[owner][i-1].bill_image,a_b[owner][i-1].product_image);
    }
}
```

1.12 References

- https://www.coursera.org/
- http://remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.6.6+commit.6c089d02.js
- https://www.w3schools.com
- www.figma.com
- www.nodejs.com
- www.youtube.com