# CS731 Course Project

Blockchain based Online Ticket Management System

| | |
|---|---|
| Group Number: | **22** |
| Project: | **Safar Saathi** |
| Mentor TA: | **Sumit Lahiri** |
| Instructor: | **Dr. Angshuman Karmakar** |

**Group Details**

Aakarsh Mittal (200002)

Kuldeep Singh Chouhan (200530)

Sandeep Kumar Bijarnia (200856)

Yash Raj Mittal (201148)

# ACKNOWLEDGEMENT

We express our heartfelt appreciation and gratitude to Prof. (Dr). Angshuman Karmakar for his invaluable guidance, mentorship, and support throughout this project at the Department of Computer Science and Engineering, Indian Institute of Technology Kanpur. His expertise, insights, and encouragement have shaped the direction and quality of our work, and we are grateful for the opportunity to learn from him.

We also extend our sincere thanks to Teaching Assistant Mr. Sumit Lahri  for his constant support and unparalleled guidance. His inputs, feedback, and assistance have been invaluable in our project's progress. Additionally, we acknowledge the support of the faculty and staff of the Department of Computer Science and Engineering at IIT Kanpur for providing us with a conducive academic environment, access to resources, and opportunities to learn and grow.

# INDEX

# 1. Introduction

Safar Saathi is an online ticket management system powered by secure and reliable blockchain technology. With its user-friendly web portal, users can easily access a wide range of transportation options and enjoy the unparalleled benefits of blockchain trust. This platform allows publishers to add transportation of various modes and publish tickets effortlessly. For consumers, purchasing tickets is a breeze with the intuitive interface. Payment is processed seamlessly via the trusted and secure MetaMask system

# 2. Tech Stack

## Truffle

Truffle provides a suite of tools and utilities that simplify the process of smart contract development, deployment, testing, and management and can easily integrate with tools like Ganache and Infura to provide a seamless development experience.

## Ganache

Ganache is a personal blockchain for Ethereum development, used to test and debug decentralized applications (DApps) before deploying them to the Ethereum network. For ease of having multiple users, we will use this in place of Infura, which has limited ETH to be added to accounts.

## ReactJS

ReactJS is used for making a user-friendly front end for the DApp.

## MetaMask

MetaMask is used to validate transactions made from the portal and receive funds for sold tickets.

# 3. Running the DApp

1. Clone the `main` branch from the GitHub Repository https://github.com/kuldeep-singh-chouhan/CS731_workspace.git

2. Change the directory to `CS731_workspace`

3. `truffle` directory contains the contracts and migrations, and `client` directory contains the front-end code.

4. Ensure truffle is installed, using command

   ```
   truffle --version
   ```

   ```
   Truffle v5.8.1 (core: 5.8.1)
   Ganache v7.7.7
   Solidity v0.5.16 (solc-js)
   Node v16.17.0
   Web3.js v1.8.2
   ```

   Download the GUI of Ganache and open it.

5. Change the directory to truffle and migrate the smart contract.

   ```
   cd truffle
   truffle migrate
   ```

6. Open a new terminal and change directory to `client` and install `npm`

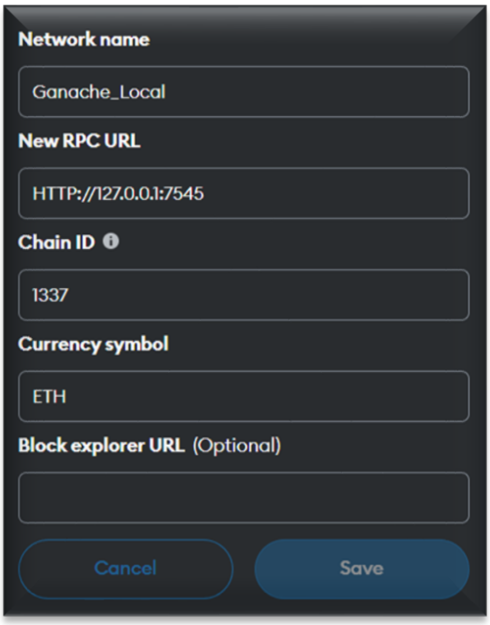
   ```
   cd ..
   cd client
   npm install --force
   ```

7. Start the frontend portal

   ```
   npm start
   ```

8. Go to the localhost `http://localhost:8080/`
9. Set up Ganache local wallet in MetaMask

a. Login to MetaMask
b. Manually setup Ganache with the given details



Network name: Ganache_Local

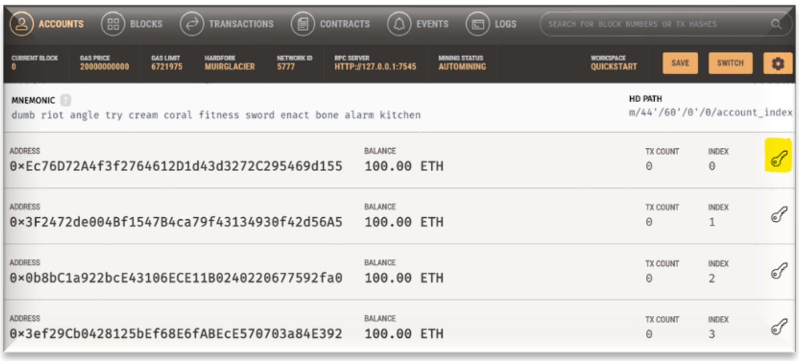
New RPC URL: HTTP://127.0.0.1:7545

Chain ID: 1337

Currency symbol: ETH

Block explorer URL (Optional):

c. Open Ganache and import an account from private key By clicking the key logo shown below

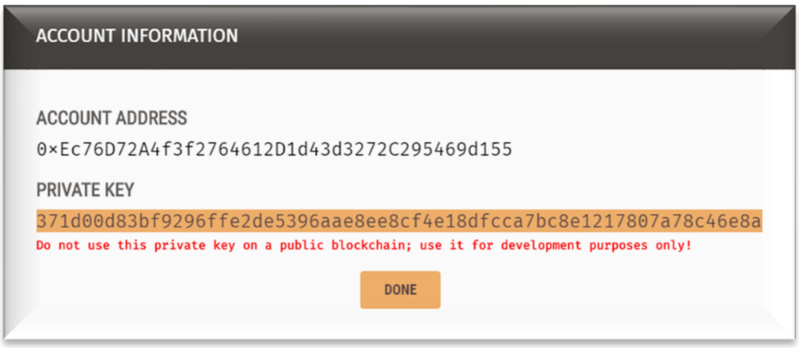


| CURRENT BLOCK | GAS PRICE | GAS LIMIT | HARDFORK | NETWORK ID | RPC SERVER | MINING STATUS | WORKSPACE |
|---|---|---|---|---|---|---|---|
| 0 | 20000000000 | 6721975 | MUIRGLACIER | 5777 | HTTP://127.0.0.1:7545 | AUTOMINING | QUICKSTART |

MNEMONIC: dumb riot angle try cream coral fitness sword enact bone alarm kitchen

HD PATH: m/44'/60'/0'/0/account_index

| ADDRESS | BALANCE | TX COUNT | INDEX |
|---|---|---|---|
| 0×Ec76D72A4f3f2764612D1d43d3272C295469d155 | 100.00 ETH | 0 | 0 |
| 0×3F2472de004Bf1547B4ca79f43134930f42d56A5 | 100.00 ETH | 0 | 1 |
| 0×0b8bC1a922bcE43106ECE11B0240220677592fa0 | 100.00 ETH | 0 | 2 |
| 0×3ef29Cb0428125bEf68E6fABEcE570703a84E392 | 100.00 ETH | 0 | 3 |



ACCOUNT INFORMATION

ACCOUNT ADDRESS

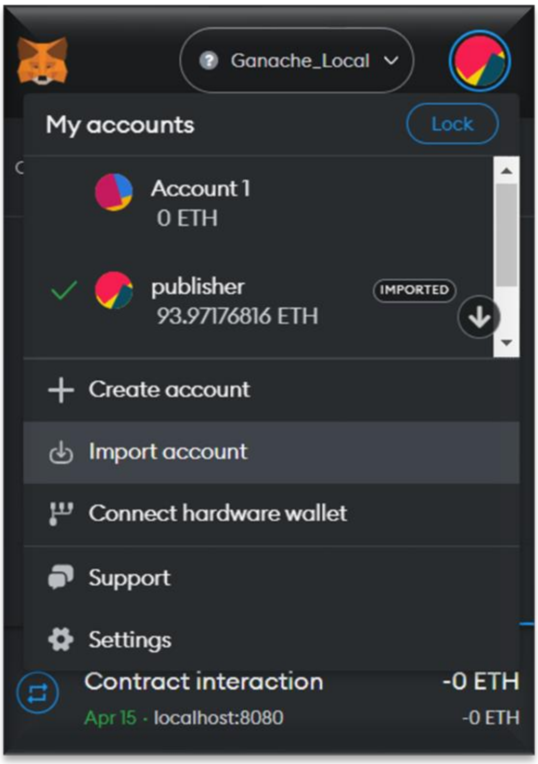
0×Ec76D72A4f3f2764612D1d43d3272C295469d155

PRIVATE KEY

371d00d83bf9296ffe2de5396aae8ee8cf4e18dfcca7bc8e1217807a78c46e8a

Do not use this private key on a public blockchain; use it for development purposes only!

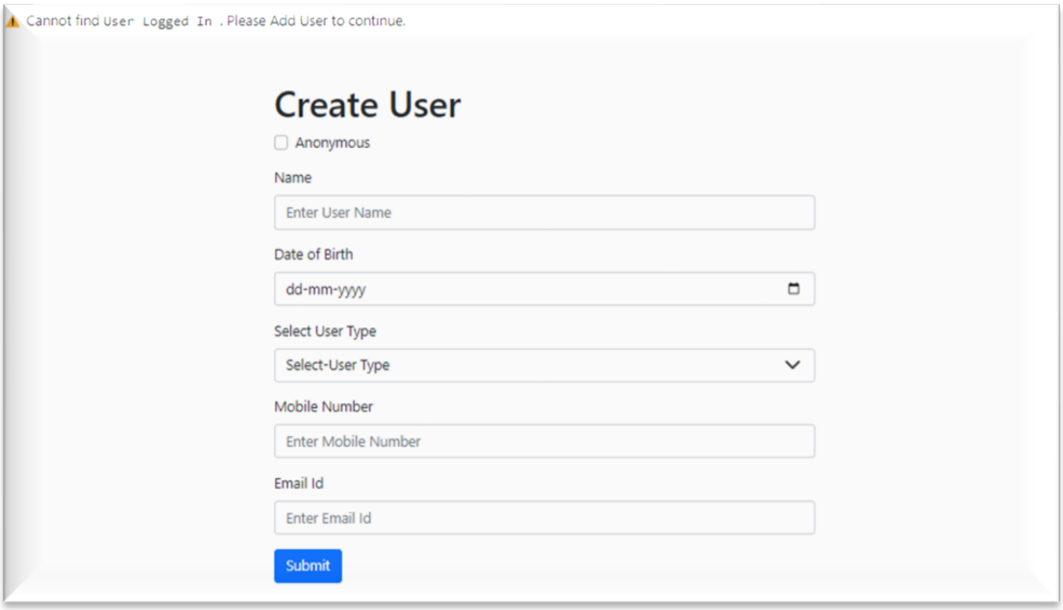
d. Import the account in MetaMask, through clicking import account and then enter and save the private key. Connect to that account.



10. You are all set when prompted to create a new user.

# 4. Using the DApp

1. When the user will open the portal first time, he will be prompted to create his profile. It can be done anonymously as well.



Cannot find User Logged In . Please Add User to continue.

Create User

☐ Anonymous

Name: Enter User Name

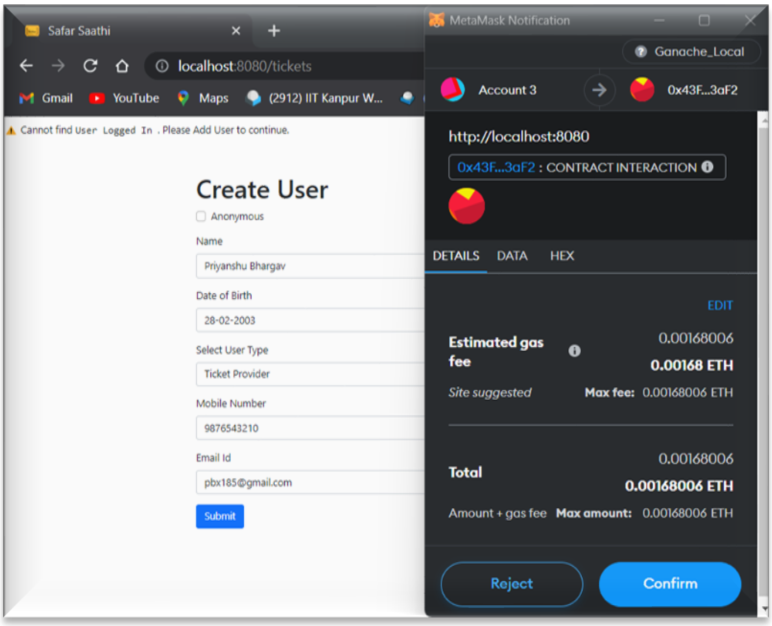
Date of Birth: dd-mm-yyyy

Select User Type: Select-User Type

Mobile Number: Enter Mobile Number

Email Id: Enter Email Id

2. Confirm the transaction from the MetaMask



Cannot find User Logged In . Please Add User to continue.

Create User

☐ Anonymous

Name: Priyanshu Bhargav

Date of Birth: 28-02-2003

Select User Type: Ticket Provider

Mobile Number: 9876543210

Email Id: pbx185@gmail.com

http://localhost:8080

0x43F...3aF2 : CONTRACT INTERACTION

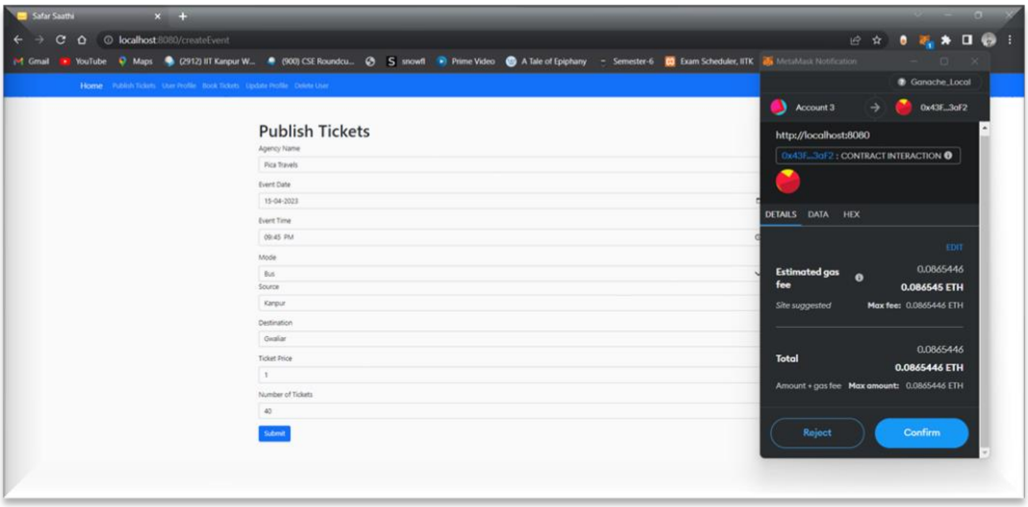
Estimated gas fee 0.00168006 0.00168 ETH

Site suggested Max fee: 0.00168006 ETH

Total 0.00168006 0.00168006 ETH

Amount + gas fee Max amount: 0.00168006 ETH

3. If chosen Ticket Provider, user can publish tickets by filling the form, submitting and confirming transaction through MetaMask.



## Publish Tickets

Agency Name: Pica Travels

Event Date: 15-04-2023

Event Time: 09:45 PM

Mode: Bus

Source: Kanpur

Destination: Gwaliar

Ticket Price: 1

Number of Tickets: 40

http://localhost:8080

0x43F...3aF2 : CONTRACT INTERACTION

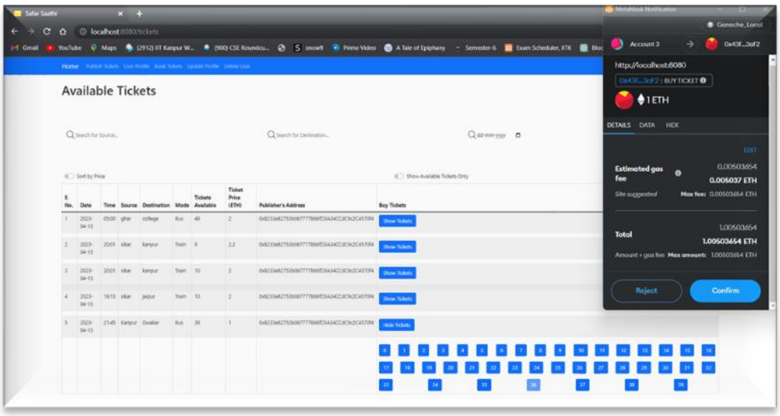
Estimated gas fee 0.0865446 0.086545 ETH

Site suggested Max fee: 0.0865446 ETH

Total 0.0865446 0.0865446 ETH

Amount + gas fee Max amount: 0.0865446 ETH

4. In User Profile, you can see your details, published events and your bought tickets. The bought tickets can be cancelled by clicking `Sell Tickets`. To book tickets you can select the transport provider and then select a seat. To delete your profile, click on `Delete User`. In `Update Profile` tab, one can update his details.



## Available Tickets

| S No. | Date | Time | Source | Destination | Mode | Tickets Available | Ticket Price (ETH) | Publisher's Address | Buy Tickets |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2023-04-15 | 05:00 | ghar | college | Bus | 49 | 2 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | Show Tickets |
| 2 | 2023-04-15 | 20:01 | sikar | kanpur | Train | 9 | 2.2 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | Show Tickets |
| 3 | 2023-04-15 | 20:01 | sikar | kanpur | Train | 10 | 2 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | Show Tickets |
| 4 | 2023-04-15 | 19:13 | sikar | jaipur | Train | 10 | 2 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | Show Tickets |
| 5 | 2023-04-15 | 21:45 | Kanpur | Gwaliar | Bus | 39 | 1 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | Hide Tickets |

http://localhost:8080

0x43F...3aF2 : BUY TICKET

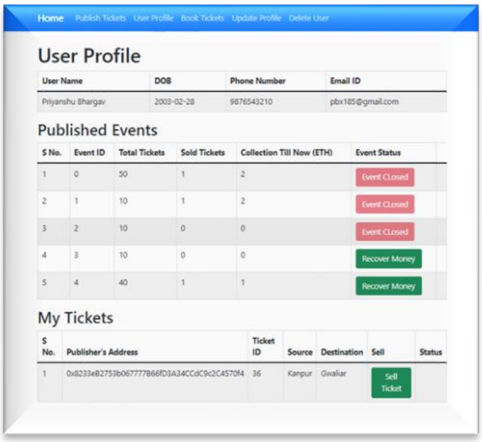
1 ETH

Estimated gas fee 0.00503654 0.005037 ETH

Site suggested Max fee: 0.00503654 ETH

Total 1.00503654 1.00503654 ETH

Amount + gas fee Max amount: 1.00503654 ETH



## User Profile

| User Name | DOB | Phone Number | Email ID |
|---|---|---|---|
| Priyanshu Bhargav | 2003-02-28 | 9876543210 | pbx185@gmail.com |

## Published Events

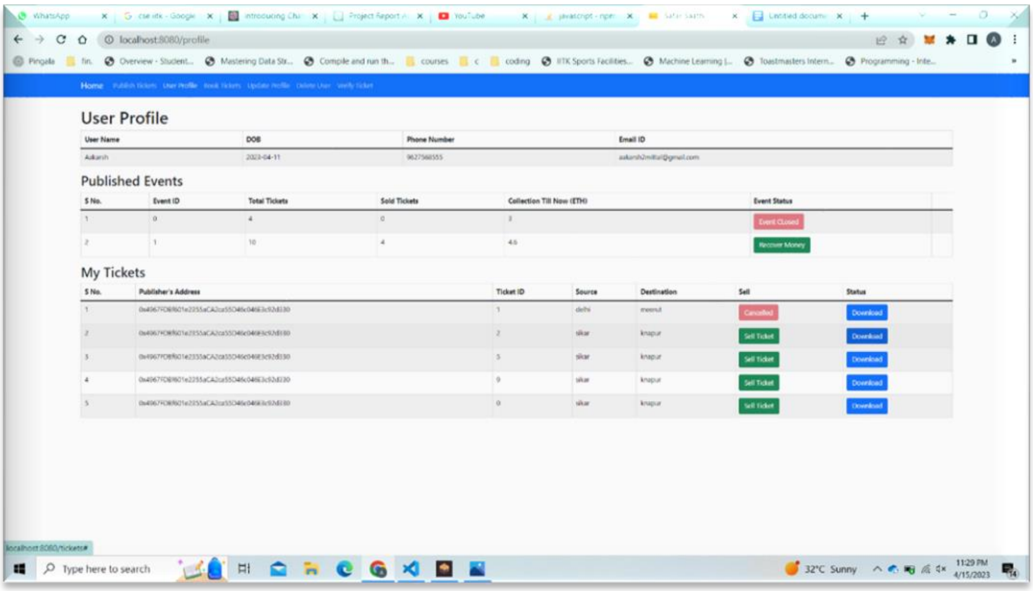
| S No. | Event ID | Total Tickets | Sold Tickets | Collection Till Now (ETH) | Event Status |
|---|---|---|---|---|---|
| 1 | 0 | 50 | 1 | 2 | Event CLosed |
| 2 | 1 | 10 | 1 | 2 | Event CLosed |
| 3 | 2 | 10 | 0 | 0 | Event CLosed |
| 4 | 3 | 10 | 0 | 0 | Recover Money |
| 5 | 4 | 40 | 1 | 1 | Recover Money |

## My Tickets

| S No. | Publisher's Address | Ticket ID | Source | Destination | Sell | Status |
|---|---|---|---|---|---|---|
| 1 | 0x8233eB2753b067777B66fD3A34CCdC9c2C4570f4 | 36 | Kanpur | Gwaliar | Sell Ticket | |

5. Similar things is done for buyer, except publishing tickets, where error will be prompted.

6. After booking the ticket, you can download your ticket from the `User Profile`.



User Profile

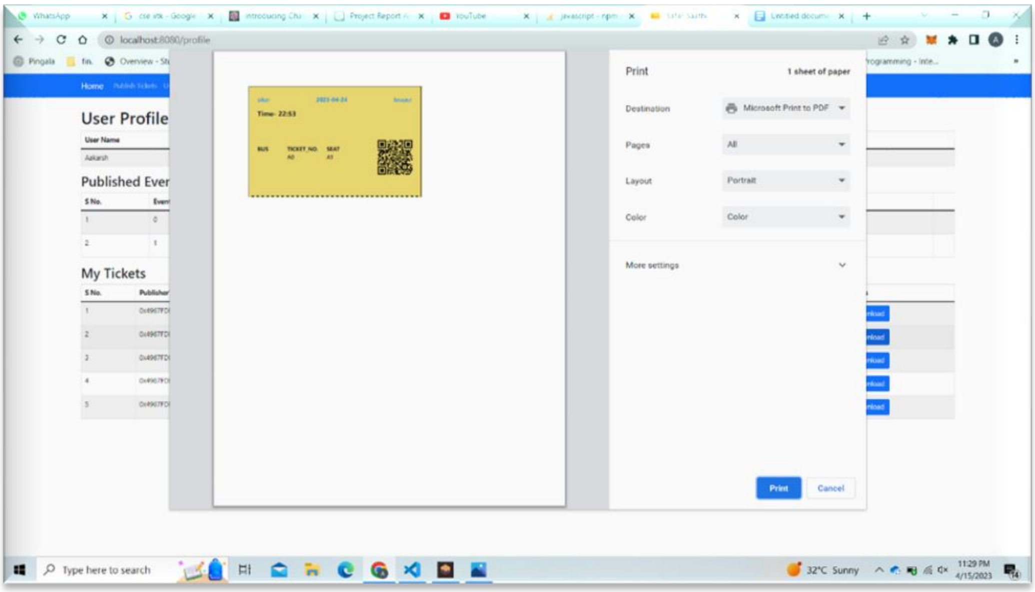
| User Name | DOB | Phone Number | Email ID |
|---|---|---|---|
| Aakarsh | 2023-04-11 | 9627568555 | aakarsh2mittal@gmail.com |

Published Events

| S No. | Event ID | Total Tickets | Sold Tickets | Collection Till Now (ETH) | Event Status |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 0 | 3 | Event Closed |
| 2 | 1 | 10 | 4 | 4.6 | Recover Money |

My Tickets

| S No. | Publisher's Address | Ticket ID | Source | Destination | Sell | Status |
|---|---|---|---|---|---|---|
| 1 | 0x4967FD8f601e2355aCA2ca55D46c046E3c92d330 | 1 | delhi | meerut | Cancelled | Download |
| 2 | 0x4967FD8f601e2355aCA2ca55D46c046E3c92d330 | 2 | sikar | knapur | Sell Ticket | Download |
| 3 | 0x4967FD8f601e2355aCA2ca55D46c046E3c92d330 | 5 | sikar | knapur | Sell Ticket | Download |
| 4 | 0x4967FD8f601e2355aCA2ca55D46c046E3c92d330 | 9 | sikar | knapur | Sell Ticket | Download |
| 5 | 0x4967FD8f601e2355aCA2ca55D46c046E3c92d330 | 0 | sikar | knapur | Sell Ticket | Download |



Print — 1 sheet of paper

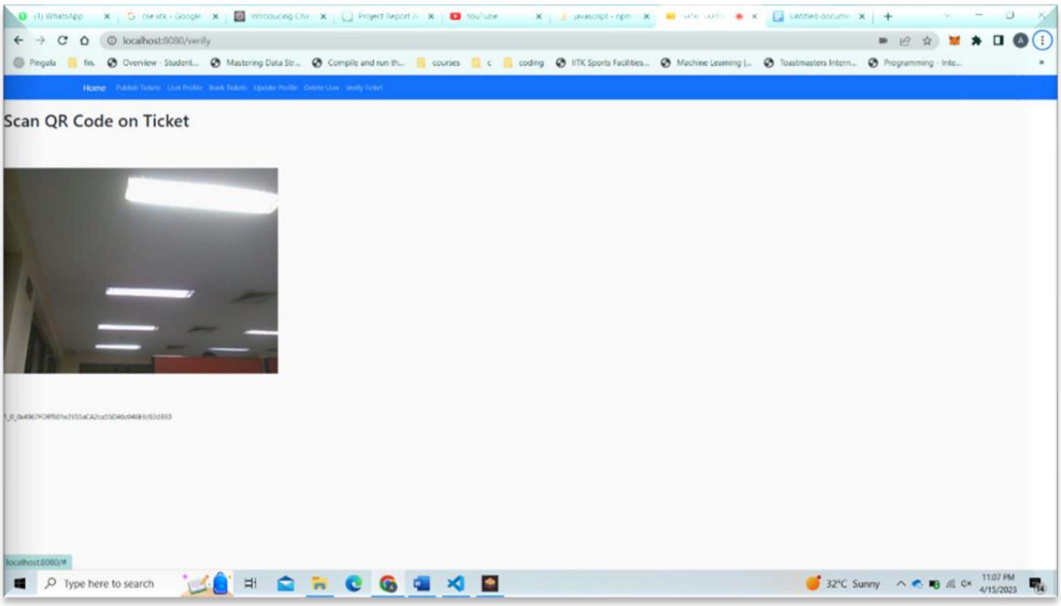
Destination: Microsoft Print to PDF

Pages: All

Layout: Portrait

Color: Color

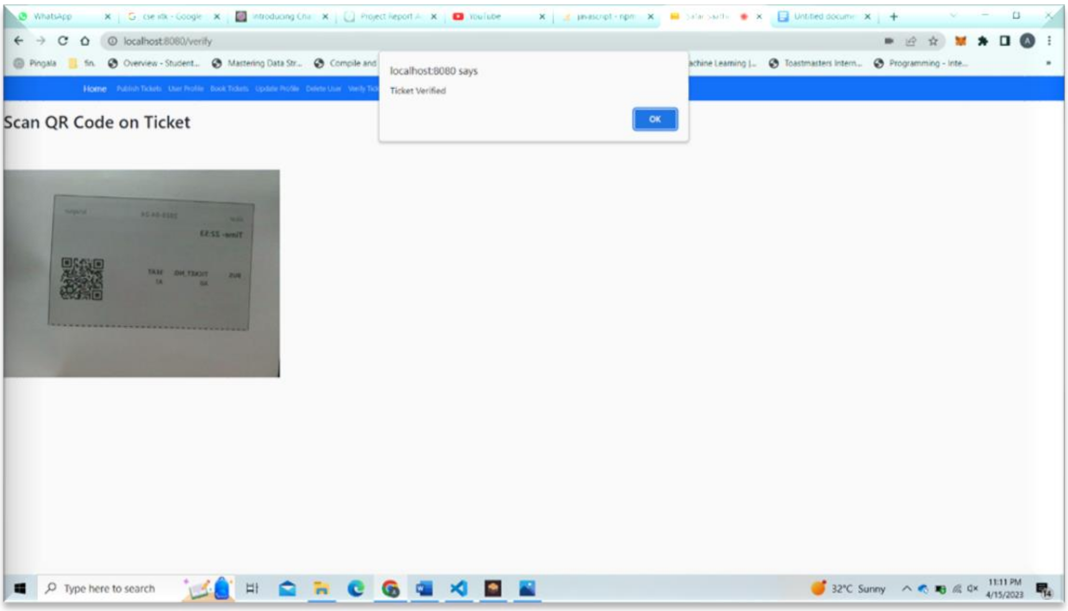
7. In `Verify Ticket,` you can verify your ticket with the help of the QR code present on the ticket.



Scan QR Code on Ticket

1_0_0x4967FO8f601e2355aCA2ca55D46c0468 3c92d330

Ticket verified notification is given as soon as the ticket's QR code is showed to the camera.



Scan QR Code on Ticket

localhost:8080 says

Ticket Verified

# 5. Smart Contract

`truffle` directory contains the smart contract in subdirectory `contracts`. There is one contract `EventFactory`, which contains struct for `User`, `Ticket` and `Event` along with functions on them. Functionalities provided are

1. `addUser`, `deleteUser`, `getUserDetails` and `UserExist` are used for user operations to add, delete, getting details of and checking existence of user respectively.
2. `createEvent`, `showAllEvents`, `showAvailableEvents` and `showUserEvents` are Event based functions to creating, showcasing all, showcasing unsold and showcasing User's published events respectively.
3. `buyTicket`, `sellTicket` and `showUserTickets` are used for buying cancelling (reselling) and showing bought tickets respectively. When a user cancels a ticket, only 60% of the ticket amount is refunded.
4. `currentPrice` dynamically changes the ticket price based on its demand and returns the current price of the ticket. The ticket price can rise upto its 1.5 times.
5. `bal` function returns the balance of a user. If the user is a ticket publisher, he can use `recoverFund` to stop accepting further ticket request and receive the collected amount through that event into his wallet.
6. User details are stored in firebase, other data are on chain.
7. For testing, we have a script in `scripts` directory. Which can be run in truffle directory by `truffle exec scripts/inc.js`

# 6. Contribution

Date: 21/03/23

Log1:

Repo link: https://github.com/kuldeep-singh-chouhan/CS731_workspace

branch {main}

| | | |
|---|---|---|
| Kuldeep Singh Chouhan | 200530 | Function to show available tickets and integrating with goerli test net and meta mask |
| Yash Raj Mittal | 201148 | Function for creating event, buying and selling tickets and integration with ganache |
| Sandeep Kumar Bijarnia | 200856 | Function for adding/deleting users and user struct |
| Aakarsh Mittal | 200002 | Struct for the tickets, event and mappings |

Date: 03/04/23

Log2:

Repo link: https://github.com/kuldeep-singh-chouhan/CS731_workspace

branch {main}

Added and updated contracts-

1. showAllEvents
2. createEvent: added source destination and can accumulate the overall collection from selling tickets of that event
3. showAvailableEvents: search by source and destination
4. currentPrice: support for dynamic pricing based on number of tickets sold
5. sellTickets: refund of cancelled tickets
6. showUserTickets

7. recoverFund: owner gets the collected amount from ticket selling and after this, the event will be closed

8. bal: shows the balance of the contract

9. Fixing errors related to sending transactions, which required gas

10. integrated ganache with metamask

11. integrated truffle with react, and implemented the previous written smart contracts on frontend

12. written scripts for testing the smart contracts hosted on ganache

| | | |
|---|---|---|
| Kuldeep Singh Chouhan | 200530 | 3,5,9,12 |
| Yash Raj Mittal | 201148 | 4, 2, 7,12 |
| Sandeep Kumar Bijarnia | 200856 | 8, 10, 11,12 |
| Aakarsh Mittal | 200002 | 1, 6 |

Date: 15/04/2023

Log3:

Repo link: https://github.com/kuldeep-singh-chouhan/CS731_workspace

branch {main}

Backend Related Updates-

1. Mapping for ticketToUser and eventToUser and related helper functions to get data in frontend.

2. Connected Firebase to store UserData and EventData.

Final/Frontend Updates-

Modified and Added Several Components on the frontend:-

3. Created Profile Component with section of Published Events with recover Fund functionality and My Tickets Componentto sell the tickets.

4. Update Profile section

5. Publish Ticket for creating new tickets only accesible to Publishers.

6. Delete User option

7. Add New User Section

8. Book Tickets section with search based filter on several Parameters to buy tickets based on EventId and TicketNumber.

9. User Can download ticket

10. Ticket Can be verified using QR code

11. Testing Script for Checking Multiple Users and Tickets

12. Project Report and Activity Log

| | | |
|---|---|---|
| Kuldeep Singh Chouhan | 200530 | 5,7,11,12 |
| Yash Raj Mittal | 201148 | 1,3,9,12 |
| Sandeep Kumar Bijarnia | 200856 | 2,8,10,12 |
| Aakarsh Mittal | 200002 | 4,6,12 |