Names of Students in the group:

1. Kehinde Akinmolayan
2. Henry Odusanya
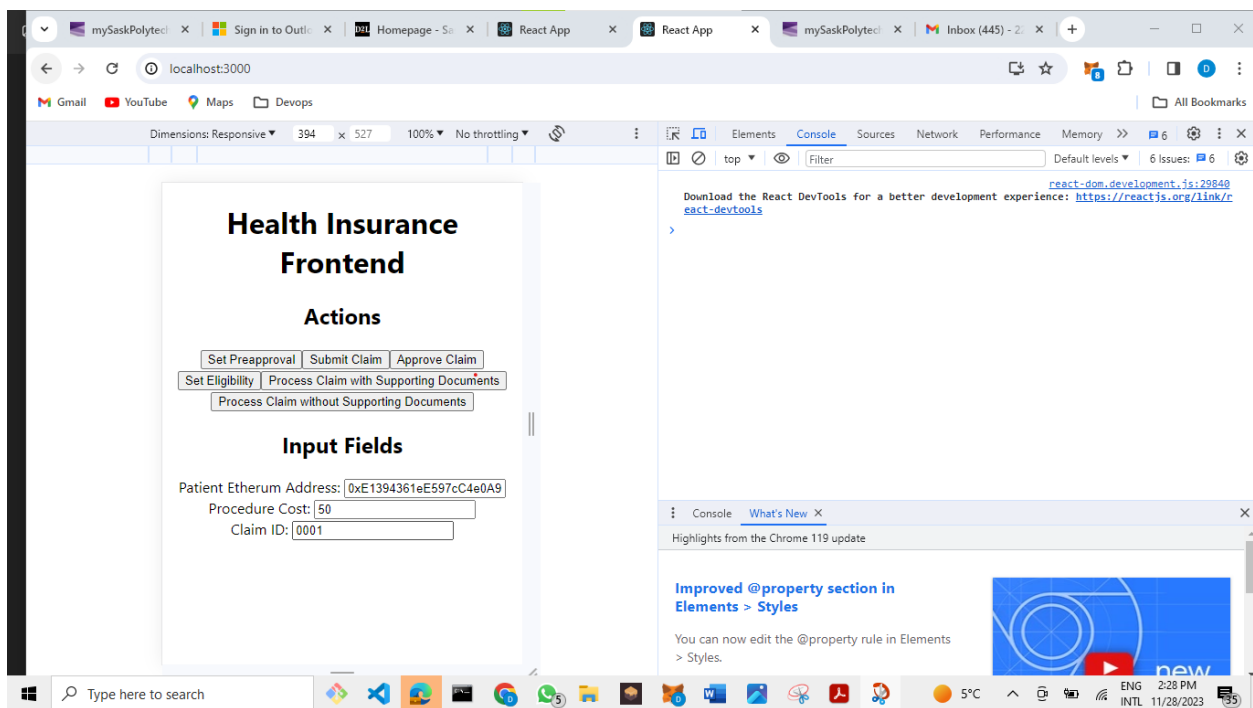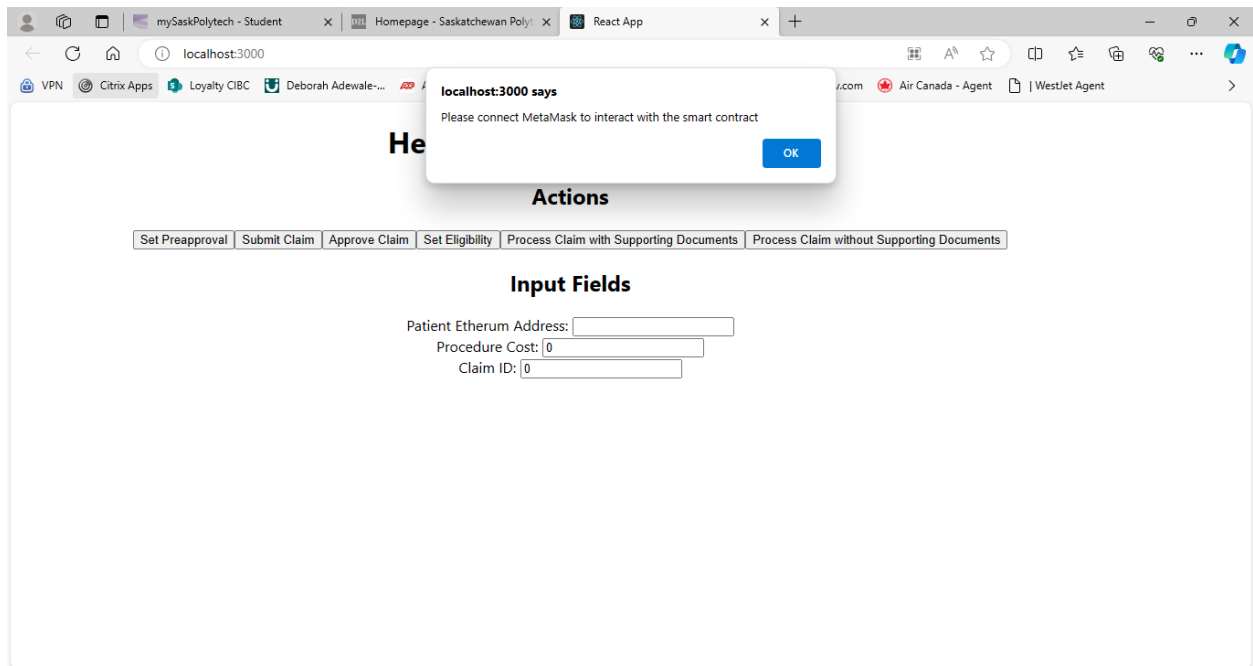3. Oluwafunmilayo Adewale-Michael

Steps to set up the project:

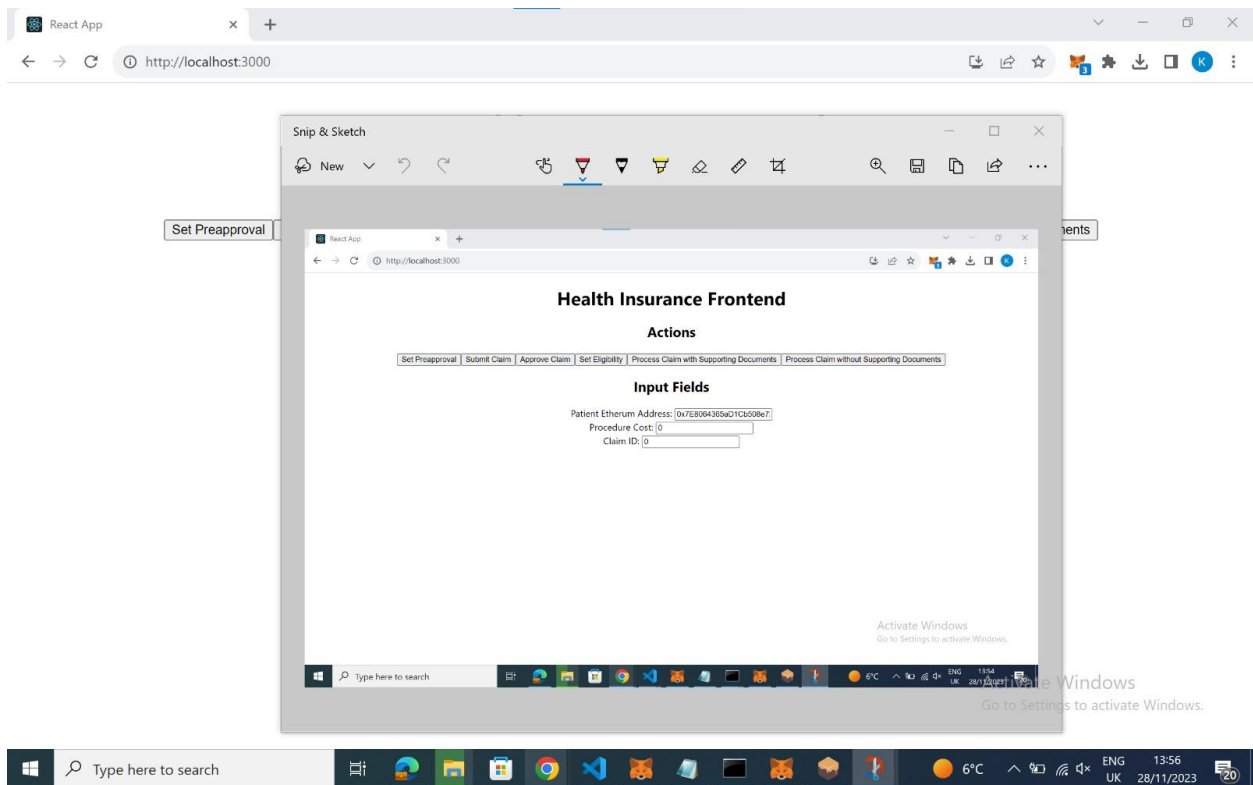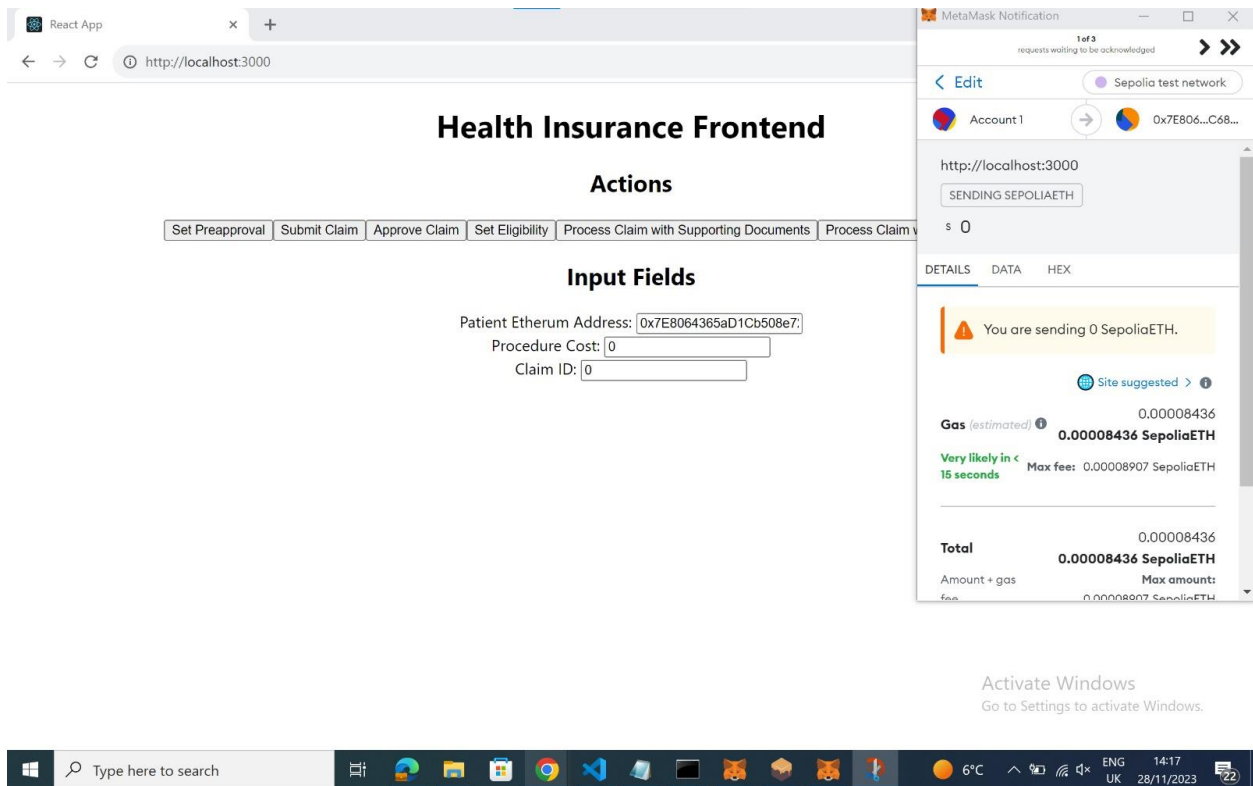Steps to Deploy the Contract on Ganache

1. Write the smart contract with solidity, and test the functionalities of the contract on remix
2. On the terminal, create a folder called **mkdir smarthealth**
3. cd into the folder;
4. type and run the command
   a. **truffle init**
   b. **truffle create contract HealthInsuranceContract**
5. with **code .** open vscode
6. copy and paste the smartcontract from remix into the HealthInsuranceContract file
7. edit truffle-config.js file
   a. change the version to version: 0.8.17
   b. change the port to 7545
   c. network to * to accept all
8. on the terminal, **truffle create migration HealthInsuranceContract**
9. in vscode update the .js migration file accordingly
10. on the terminal, **truffle deploy**

Move out of the above folder and create a new folder for the frontend

1. npx create-react-app my-app
2. npm install web3
3. create a file HealthInsuranceContract.json inside the src folder
4. copy the .json truffle artifact from the build/contract/HealthInsuranceContract.json
5. paste in the file created in step3
6. Edit the app.js file
   a. Add contract abi and contract address
   b. Added our functions to interact with metamask
   c. Included the html code for the frontend
7. npm start

Screenshots of interaction with the smart contract:

localhost:3000

He

Actions

Set Preapproval | Submit Claim | Approve Claim | Set Eligibility | Process Claim with Supporting Documents | Process Claim without Supporting Documents

Input Fields

Patient Etherum Address:
Procedure Cost: 0
Claim ID: 0

---

localhost:3000

Health Insurance Frontend

Actions

Set Preapproval | Submit Claim | Approve Claim
Set Eligibility | Process Claim with Supporting Documents
Process Claim without Supporting Documents

Input Fields

Patient Etherum Address: 0xE1394361eE597cC4e0A9
Procedure Cost: 50
Claim ID: 0001

Elements | Console | Sources | Network | Performance | Memory

top | Filter | Default levels | 6 Issues:

Download the React DevTools for a better development experience: https://reactjs.org/link/react-devtools

Console | What's New

Highlights from the Chrome 119 update

Improved @property section in Elements > Styles

You can now edit the @property rule in Elements > Styles.

ENG
INTL 2:28 PM
11/28/2023

Health Insurance Frontend

Actions

Set Preapproval | Submit Claim | Approve Claim | Set Eligibility | Process Claim with Supporting Documents | Process Claim w

Input Fields

Patient Etherum Address: 0x7E8064365aD1Cb508e7:
Procedure Cost: 0
Claim ID: 0

MetaMask Notification

1 of 3
requests waiting to be acknowledged

< Edit — Sepolia test network

Account 1 → 0x7E806...C68...

http://localhost:3000

SENDING SEPOLIAETH

$ 0

DETAILS | DATA | HEX

⚠ You are sending 0 SepoliaETH.

🌐 Site suggested >

Gas (estimated) ⓘ 0.00008436
0.00008436 SepoliaETH

Very likely in < 15 seconds — Max fee: 0.00008907 SepoliaETH

Total — 0.00008436
0.00008436 SepoliaETH

Amount + gas fee — Max amount: 0.00008907 SepoliaETH

Snip & Sketch

New

Set Preapproval

Variable types:

Patient Ethereum Address – Address

ClaimId – uint256

Procedure cost – uint256

The expected result:

Deploying this contract creates an instance of the health insurance contract on the Ethereum blockchain.

Users can interact with the contract to set preapproval, submit claims, approve claims, set eligibility, and process claims.

The outcome varies depending on if the user is the patient or the insurance company.

The patient submits a claim on the smartcontract without the encumberances of a traditional contract, the claim is processed and approved by the insurance company, the cost of the submitclaim transaction is billed on the metamask account of the patient.