

The screenshot shows a Windows desktop environment with a browser window and a code editor side-by-side.

Browser Window: The browser is running on localhost:3000. It displays a dark-themed web application with a logo and navigation links for "Services", "Transactions", and "Footer".

Code Editor (VS Code):

- File Path:** client > src > components > Welcome.jsx
- Code Content:** The code is a React component named Welcome. It imports several icons from react-icons and defines a functional component that returns a flex container with two inner divs. The component is then exported.

```
1 import { AiFillPlayCircle } from "react-icons/ai";
2 import { SiEthereum } from "react-icons/si";
3 import { BsInfoCircle } from "react-icons/bs";
4
5 import { Loader } from ".";
6
7
8
9 const Welcome = () => [
10   return (
11     <div className="flex w-full justify-center items-center">
12       <div className="flex mf:flex-row flex-col items-start justify-between md:p-20 py-12 px-4">
13         </div>
14       </div>
15     );
16   }
17
18 export default Welcome;
19
20
```

- Terminal:** A terminal tab is open in the VS Code interface, showing the command > TERMINAL.
- Status Bar:** The status bar at the bottom shows the current file is Welcome.jsx - web3.0 - ... with 15 lines and 7 columns, and the encoding is UTF-8.
- System Tray:** The system tray shows the date and time as 13-06-2022 04:55 PM.

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View:

- The browser title bar shows three tabs: "install ubu" (closed), "Install Ubu" (closed), and "Vite App".
- The address bar displays "localhost:3000".
- The page content features a dark blue header with the "KRYPT" logo. Below it is a large white box containing the text "Send Crypto across the world" and "Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto.", both highlighted with a red rectangular border.
- At the bottom of the page are links for "Services", "Transactions", and "Footer".

Code Editor View:

- The code editor shows the file "Welcome.jsx" with the following content:

```
client > src > components > Welcome.jsx > Welcome
```

```
12   items-center">
13     <div className="flex mf:flex-row
14       flex-col items-start justify-between
15       md:p-20 py-12 px-4">
16       <div className="flex flex-1
17         justify-start items-start flex-col
18         mf:mr-10">
19         <h1 className="text-3xl
20           sm:text-5xl text-white
21             text-gradient py-1">
22             Send Crypto <br /> across the
23             world
24           </h1>
25           <p className="text-left mt-5
26             text-white font-light md:w-9/12
27             w-11/12 text-base">
28             Explore the crypto world. Buy and
29             sell cryptocurrencies easily on
30             Krypto.
31           </p>
32         </div>
33       </div>
34     </div>
```

- The code editor interface includes standard tools like search, file operations, and a terminal tab at the bottom.

The image shows a dual-screen setup. On the left screen, a web browser displays a dark-themed website for 'KRYPT'. The page features a logo with a stylized 'K' and the word 'KRYPT'. Below the logo, a large heading reads 'Send Crypto across the world'. A subtext below it says 'Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto.' A prominent blue button with white text and a circular icon says 'Connect Wallet'. On the right screen, a code editor (VS Code) shows the source code for 'Welcome.jsx' from a 'client' directory. The code includes a paragraph about buying and selling cryptocurrencies on Krypto, followed by a styled button component. The code editor interface shows tabs for 'PROBLEMS', 'OUTPUT', and 'TERMINAL', along with a terminal window at the bottom.

```
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
```

```
and sell cryptocurrencies
easily on Krypto.
</p>

<button
  type="button"
  onClick={connectWallet}
  className="flex flex-row
justify-center items-center
my-5 bg-[#2952e3] p-3
rounded-full cursor-pointer
hover:bg-[#2546bd]"
>
  <AiFillPlayCircle
  className="text-white mr-2"
  >
    <p className="text-white
text-base font-semibold">
      Connect Wallet
    </p>
  </AiFillPlayCircle>
</button>
</div>
</div>
```

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View:

- Address bar: `localhost:3000`
- Page content:
 - # Send Crypto across the world
 - Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto.
 - [Connect Wallet](#)
 - Reliability
 - Services
Transactions
Footer

Code Editor View:

- File: Welcome.jsx
- Path: client > src > components > Welcome.jsx
- Code snippet:

```
33     <AiFillPlayCircle
34     className="text-white mr-2"
35     <p className="text-white
36         text-base font-semibold">
37         Connect Wallet
38     </p>
39
40     <div className="grid sm:grid-cols-3
41         grid-cols-2 w-full mt-10">
42         <div className={CommonStyles}>
43             Reliability
44         </div>
45     </div>
46
47
48
49
50
51
52         </div>
53     </div>
54     </>;
55 );
```
- Tooltips: "Select Indentation" is visible near the bottom right of the code area.
- Status bar: Shows file path (`Welcome.jsx - web3.0 - ...`), line number (40), column number (13), character count (173 selected), spaces (4), encoding (UTF-8), file type (JavaScript React), and date/time (06:27 PM 13-06-2022).

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View:

- The browser title bar shows three tabs: "install ubu", "Install Ubu", and "Vite App".
- The address bar displays "localhost:3000".
- The page content includes:
 - A header with the text "Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto."
 - A blue button labeled "Connect Wallet".
 - A grid table with four cells:

Reliability	Security
Ethereum	Web 3.0
Low Fees	Blockchain
 - Links at the bottom: "Services" and "Transactions".

Code Editor View:

- The code editor window is titled "Welcome.jsx".
- The file path is "client > src > components > Welcome.jsx".
- The code is written in JSX and uses styled-components. It defines a grid layout with three columns and two rows, containing elements for Reliability, Security, Ethereum, Web 3.0, Low Fees, and Blockchain.

```
39
40
41 <div className="grid sm:grid-cols-3 grid-cols-2 w-full mt-10">
42
43   <div className={`rounded-tl-2xl ${CommonStyles}>
44     Reliability
45   </div>
46   <div className={CommonStyles}>
47     Security</div>
48
49   <div className={`${sm:rounded-tr-2xl ${CommonStyles}>
50     Ethereum
51   </div>
52
53   <div className={`${sm:rounded-bl-2xl ${CommonStyles}>
54     Web 3.0
55   </div>
56
57   <div className={CommonStyles}>Low Fees</div>
58   <div className={`rounded-br-2xl ${CommonStyles}>
59     Blockchain
60   </div>
```

- The status bar at the bottom of the code editor shows: Line 61, Col 13, Spaces: 4, CRLF, {}, JavaScript React, Go Live, and the date/time: 06:45 PM 13-06-2022.

install ubu x | Install Ubu x | Vite App x

localhost:3000

Google Mails YouTube Udemy HL Network... Trade Other bookmarks

Send Crypto across the world

Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto.

Connect Wallet

Reliability	Security	Ethereum
Web 3.0	Low Fees	Blockchain



File Edit Selection View ... Welcome.jsx - web3.0 ...

Welcome.jsx x

client > src > components > Welcome.jsx > Welcome

```
60   </div>
61   </div>
62
63
64
65   <div className="flex flex-col flex-1 items-center justify-start w-full mf:mt-0 mt-10">
66     <div className="p-3 flex justify-end items-start flex-col rounded-xl h-40 sm:w-72 w-full my-5 eth-card .white-glassmorphism ">
67
68   </div>
69
70   </div>
71
72
73
74
75
76
77
78
79
80
81   </div>
82   </div>
83 }
84
85
86 export default Welcome;
```

Services

0 0 Ln 73, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript React Go Live 06:59 PM 13-06-2022 2

The screenshot shows a dual-monitor setup. On the left monitor, a web browser displays a landing page for a crypto exchange named 'Krypto'. The page features a large header 'Send Crypto across the world', a sub-header 'Explore the crypto world. Buy and sell cryptocurrencies easily on Krypto.', and a 'Connect Wallet' button. Below this is a grid comparing 'Reliability', 'Security', and 'Ethereum' across three categories: 'Web 3.0', 'Low Fees', and 'Blockchain'. A callout box highlights the 'Ethereum' section of the grid. At the bottom, there's a purple button labeled 'Address' and 'Ethereum'.

On the right monitor, a code editor in Visual Studio Code shows the React component 'Welcome.jsx'. The code defines a card for Ethereum, which includes an address and an Ethereum icon. Blue arrows point from specific UI elements on the left to the corresponding code on the right, illustrating the mapping between the user interface and the underlying code.

```
client > src > components > Welcome.jsx > Welcome
  <div className="p-3 flex justify-end items-start flex-col rounded-xl h-40 sm:w-72 w-full my-5 eth-card .white-glassmorphism">
    <div className="flex justify-between flex-col w-full h-full">
      <div className="flex justify-between items-start">
        <div className="w-10 h-10 rounded-full border-2 border-white flex justify-center items-center">
          <SiEthereum fontSize={21} color="#fff" />
        </div>
        <BsInfoCircle fontSize={17} color="#fff" />
      </div>
      <div>
        <p className="text-white font-light text-sm">Address</p>
        <p className="text-white font-semibold text-lg mt-1">Ethereum</p>
      </div>
    </div>
  </div>
```

The image shows a split-screen development environment. On the left is a browser window titled "Vite App" displaying a dark-themed web application. The app has a top navigation bar with links to "Google", "Mails", "YouTube", "Udemy HL Network...", "Trade", and "Other bookmarks". Below the bar is a grid with four cells: "Ethereum" (top-left), "Web 3.0" (top-right), "Low Fees" (bottom-left), and "Blockchain" (bottom-right). A large button labeled "Address Ethereum" is visible, along with a placeholder input field "Address To". On the right is a code editor window titled "Welcome.jsx" showing the corresponding JSX code. The code defines a component structure with nested divs and an input field for sending Ethereum. Blue arrows point from the "Address Ethereum" button and the "Address To" input field in the browser to the respective lines of code in the editor.

```
client > src > components > Welcome.jsx > Welcome
  84
  85 <p className="text-white font-semibold text-lg mt-1">
  86   Ethereum
  87 </p>
  88 </div>
  89 </div>
  90
  91
  92
  93
  94
  95
  96
  97
  98
  99
  100
  101
  102
  103
  104
```

```
</p>
<p className="text-white font-semibold text-lg mt-1">
  Ethereum
</p>
</div>
</div>

<div className="p-5 sm:w-96 w-full flex flex-col justify-start items-center blue-glassmorphism">
  <Input placeholder="Address To" name="addressTo" type="text" handleChange={()=>{}} />
</div>
</div>
```

Vite App

localhost:3000

Ethereum

Web 3.0

Low Fees

Blockchain

Address

Ethereum

Address To

Services

Transactions

Footer

Welcome.jsx

```
const CommonStyles = "min-h-[70px] sm:px-0 px-2 sm:min-w-[120px] flex justify-center items-center border-[0.5px] border-gray-400 text-sm font-light text-white";
```

```
const Input = ({ placeholder, name, type, value, handleChange }) => (
  <input
    placeholder={placeholder}
    type={type}
    step="0.0001"
    value={value}
    onChange={({ e }) => handleChange(e, name)}
    className="my-2 w-full rounded-sm p-2 outline-none bg-transparent text-white border-none text-sm white-glassmorphism"
  />
);
```

```
const Welcome = () => {
  const connectWallet =()=>{}

  return (
    <div className="flex w-full justify-center items-center">
      <div className="flex mf:flex-row flex-col items-start justify-between md:p-20 py-12">
```

Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript React ⚡ Go Live 🔍 11:11 PM 13-06-2022 📈 2

The image shows a split-screen development environment. On the left is a web browser window titled "Vite App" displaying a dark-themed UI for sending Ethereum transactions. The UI includes fields for "Address To", "Amount (ETH)", "Keyword (Gif)", and "Enter Message". On the right is a code editor showing the `Welcome.jsx` file from a React application. The code defines a form component with four input fields, each with a handleChange prop that calls a function with an empty array as its argument. Blue arrows point from the input fields in the browser to their corresponding JSX code in the editor.

```
client > src > components > Welcome.jsx > Welcome
95     </div>
96
97
98     <div className="p-5 sm:w-96 w-full flex
99         flex-col justify-start items-center
100        blue-glassmorphism">
101            <Input placeholder="Address To"
102                name="addressTo" type="text" handleChange={()
103                    =>{}}
104                />
105
106            <Input placeholder="Amount (ETH)"
107                name="amount" type="number" handleChange={()
108                    =>{}}
109                />
110
111            <Input placeholder="Keyword (Gif)"
112                name="keyword" type="text" handleChange={()
113                    =>{}}
114                />
115
116            <Input placeholder="Enter Message"
117                name="message" type="text" handleChange={()
118                    =>{}}
119                />
120
121        </div>
122    </div>
```

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View: The browser window displays a Vite App titled "Vite App" at the URL "localhost:3000". The page has a dark theme with a gradient header. It features a "Address" section with an "Ethereum" button. Below this are input fields for "Address To", "Amount (ETH)", "Keyword (Gif)", and "Enter Message". A large blue "Send now" button is at the bottom.

Code Editor View: The code editor shows the file "Welcome.jsx" in a dark-themed interface. The code defines a component "Welcome" which contains a "div" with a specific class and a "button" with an "onClick" handler. Two blue arrows point from the "Send now" button in the browser to the "onClick={handleSubmit}" line in the code editor.

```
File Edit Selection View ... Welcome.jsx - web3.0 ...
client > src > components > Welcome.jsx > Welcome
=>{()} />
109
110
111 <div className="h-[1px] w-full bg-gray-400 my-2" />
112
113 {false? (
114 <Loader /> )
115 :(
116 <button
117 type="button"
118 onClick={handleSubmit}
119 className="text-white w-full mt-2 border-[1px] p-2 border-[#3d4f7c]
120 border-[#3d4f7c] rounded-full cursor-pointer"
121 >
122 <span>Send now</span>
123 </button>
124 )
125 </div>
126 </div>
127 </div>
128
129
130
131
132
```

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View: The browser displays a web application titled "Vite App" at the URL "localhost:3000". The page has a gradient header with the text "Address" and "Ethereum". Below the header is a dark blue form with four input fields: "Address To", "Amount (ETH)", "Keyword (Gif)", and "Enter Message". A red circular icon is visible near the bottom left of the form area.

Code Editor View: The code editor shows the file "Welcome.jsx" with line numbers from 107 to 127. A blue arrow points from the "Enter Message" input field in the browser to the line of code at line 113, which contains the JSX for a button component.

```
client > src > components > Welcome.jsx > Welcome
  name="keyword" type="text" handleChange={()=>{}} />
  <Input placeholder="Enter Message" name="message" type="text" handleChange={()=>{}} />
  <div className="h-[1px] w-full bg-gray-400 my-2" />
  {true? [<Loader />] : (
    <button
      type="button"
      onClick={handleSubmit}
      className="text-white w-full mt-2 border-[1px] p-2 border-[#3d4f7c] !hover:bg-[#3d4f7c] rounded-full cursor-pointer"
    > Send now
    </button>
  )}
</div>
```

The screenshot shows a development environment with a browser window and a code editor side-by-side.

Browser View: The browser window displays a Vite App titled "Vite App" at "localhost:3000". The page content includes a header with "Address" and "Ethereum" buttons, followed by input fields for "Address To", "Amount (ETH)", "Keyword (Gif)", and "Enter Message". Below these fields is a large dark blue button with a red circular outline. The URL bar shows "localhost:3000".

Code Editor View: The code editor displays the file "Loader.jsx" located at "client > src > components > Loader.jsx". The code defines a functional component "Loader" that returns a styled div containing a spin icon. The code is as follows:

```
1 const Loader = () => (
2   <div className="flex justify-center items-center py-3">
3     <div className="animate-spin rounded-full h-32 w-32 border-b-2 border-red-700" />
4   </div>
5 );
6
7 export default Loader;
```

The code editor interface includes a sidebar with icons for file operations, a status bar at the bottom showing "Ln 8, Col 3", "Spaces: 4", "UTF-8", "CRLF", and "JavaScript React", and a system tray at the bottom right showing the date and time.

File Edit Selection View Go Run Terminal Help deploy.js - web3.0 - Visual Studio Code

EXPLORER ... JS hardhat.config.js Transaction.sol ● JS deploy.js X

WEB3.0
client
smart_contract
artifacts
cache
contracts
Transaction.sol
node_modules
scripts
JS deploy.js
test
JS sample-test.js
.gitignore
JS hardhat.config.js
{ package-lock.json
{ package.json
 README.md
 web3.0.pdf

```
smart_contract > scripts > JS deploy.js > runMain
1 const main = async () => {
2   const Transactions = await hre.ethers.getContractFactory("Transactions");
3   const transactions = await Transactions.deploy();
4
5   await transactions.deployed();
6
7   console.log("Transactions address: ", transactions.address);
8 }
9
10 const runMain = async () => {
11   try {
12     await main();
13     process.exit(0);
14   } catch (error) {
15     console.error(error);
16     process.exit(1);
17   }
18 }
19
20 runMain();
```

Ln 17, Col 4 Spaces: 2 UTF-8 LF {} JavaScript ⚡ Go Live 🔍 03:52 PM 14-06-2022

YouTube (679) Build and Deploy a Modern | MetaMask | Goerli Faucet | Alchemy

dashboard.alchemyapi.io/apps/0odrqrzucw6g5r6

Google Mails YouTube Udemy HL Network... Trade UpGrad AQUA fight Again ! GitHub Repo मराठी संकेतस्थळ NETR Passion Other bookmarks

alchemy Ethereum + L2 Dashboard Apps Explorer Composer Mempool Notify Enhanced APIs Upgrade

Blockchain smart_contract Goerli

VIEW KEY GET TEST ETH ADD TO WALLET EDIT APP

COMPUTE UNITS / SEC (LAST MIN) 0 API KEY I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW Copy

HTTPS https://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW Copy

CONC. REQUESTS (LAST 1H) 0 WEBSOCKETS wss://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW Copy

THROUGHPUT LIMITED % (LAST 24 H) 0% (0)

INVALID REQUESTS (LAST 24 H) 0

Get Support

The screenshot shows the Alchemy dashboard interface for a Goerli smart contract. At the top, there are tabs for Dashboard, Apps (selected), Explorer, Composer, Mempool, Notify, Enhanced APIs, and Upgrade. Below the tabs, there's a section for the smart contract named 'Blockchain smart_contract'. It shows the Goerli network icon and several metrics: 'COMPUTE UNITS / SEC (LAST MIN)' at 0, 'API KEY' with the value 'I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW' and a 'Copy' button, 'HTTPS' with the URL 'https://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW' and a 'Copy' button, 'CONC. REQUESTS (LAST 1H)' at 0, 'WEBSOCKETS' with the URL 'wss://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW' and a 'Copy' button, 'THROUGHPUT LIMITED % (LAST 24 H)' at 0%, and 'INVALID REQUESTS (LAST 24 H)' at 0. There's also a 'VIEW KEY' button and links for 'GET TEST ETH', 'ADD TO WALLET', and 'EDIT APP'. A 'Get Support' button is located at the bottom right. The browser taskbar at the bottom shows various icons and the system tray indicates the date and time as 10:27 AM on 15-06-2022.

YouTube (679) Build and Deploy a Modern x MetaMask x Goerli Faucet x Alchemy x +

MetaMask | chrome-extension://nkbihfbeogaeaoehlefknkodbefgpgknn/home.html#

Google Mails YouTube Udemy HL Network... Trade UpGrad AQUA fight Again ! GitHub Repo मराठी संकेतस्थळ NETR Passion » Other bookmarks

METAMASK

Rinkeby Test Network

Account 1

0xe5DEd5C8B695c7C0892340bb694e
1280935599f6

Assets

17.6497 ETH

View on Etherscan

Export Private Key

Activity >

Don't see your token? Import tokens

Need help? Contact MetaMask Support

10:36 AM 15-06-2022

The screenshot shows the MetaMask extension running in a browser. The main window displays account information for 'Account 1' with a QR code and a public key. Below this, there's a section for 'Assets' showing 17.6497 ETH. Two buttons are visible: 'View on Etherscan' and 'Export Private Key'. At the bottom, there are links for 'Import tokens' and 'Need help? Contact MetaMask Support'. The browser's address bar shows the URL of the MetaMask extension's home page. The taskbar at the bottom includes icons for File Explorer, Google Chrome, and other Windows applications.

YouTube (679) Build and Deploy a Modern x MetaMask x Goerli Faucet x Alchemy x +

MetaMask | chrome-extension://nkbihfbeogaeaoehlefknkodbefgpgknn/home.html#

Google Mails YouTube Udemy HL Network... Trade UpGrad AQUA fight Again ! GitHub Repo मराठी संकेतस्थळ NETR Passion » Other bookmarks

METAMASK

Rinkeby Test Network

Account 1

0xe5DEd5C8B695c7C0892340bb...

Show Private Keys

This is your private key (click to copy)

e3fdbac51a607f34603c8ea8f9410
3f399b9d0be0bfe0cebf356911227
1b11d

Warning: Never disclose this key. Anyone with your private keys can steal any assets held in your account.

Done

Assets

17.6497 ETH

Activity >

Need help? Contact [MetaMask Support](#)

10:37 AM 15-06-2022

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a project structure under "WEB3.0".
 - client
 - smart_contract
 - artifacts
 - cache
 - contracts
 - Transaction.sol
 - node_modules
 - scripts
 - deploy.js
 - test
 - sample-test.js
 - .gitignore
 - hardhat.config.js (selected)
 - package-lock.json
 - package.json
 - README.md
 - package-lock.json
 - package.json
 - web3.0.pdf

- EDITOR**: The "hardhat.config.js" file is open.

```
// https://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW
require("@nomiclabs/hardhat-waffle");
module.exports = {
  solidity: "0.8.4",
  networks: {
    goerli: {
      url:'https://eth-goerli.alchemyapi.io/v2/I64BCQr26iVr6Yq5mSU9MN0Fi2UeEkXW',
      accounts:[ 'e3fdbac51a607f34603c8ea8f94103f399b9d0be0bfe0cebfc3569112271b11d' ]
    }
  }
};
```
- STATUS BAR**: Shows tabs for "hardhat.config.js", "Transaction.sol", and "deploy.js".

File Edit Selection View Go Run Terminal Help Transactions.sol - web3.0 - Visual Studio Code

EXPLORER ... JS hardhat.config.js Transactions.sol X JS deploy.js

WEB3.0 smart_contract > contracts > Transactions.sol

```
event Transfer(address from, address receiver, uint amount, string message, uint256 timestamp, string keyword);

struct TransferStruct {
    address sender;
    address receiver;
    uint amount;
    string message;
    uint256 timestamp;
    string keyword;
}

TransferStruct[] transactions;
```

PROBLEMS OUTPUT TERMINAL

TERMINAL

```
PS C:\Users\CS\OneDrive\Desktop\web3.0\smart_contract> npx hardhat run scripts/deploy.js --network goerli
>>
Compiled 1 Solidity file successfully
Transactions address: 0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2
PS C:\Users\CS\OneDrive\Desktop\web3.0\smart_contract>
```

Ln 6, Col 17 Spaces: 4 UTF-8 CRLF Solidity Go Live

File Edit Selection View Go Run Terminal Help constants.js - web3.0 - Visual Studio Code

EXPLORER ... JS hardhat.config.js Transactions.sol JS constants.js X JS deploy.js

WEB3.0 client > src > utils > JS constants.js > [?] contractAddress

1 export const contractAddress = '0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2';

App.css
App.jsx
favicon.svg
index.css
logo.svg
main.jsx
.gitignore
index.html
{ package-lock.json
{ package.json
JS postcss.config.js
JS tailwind.config.js
JS vite.config.js
smart_contract artifacts cache contracts

PROBLEMS OUTPUT TERMINAL

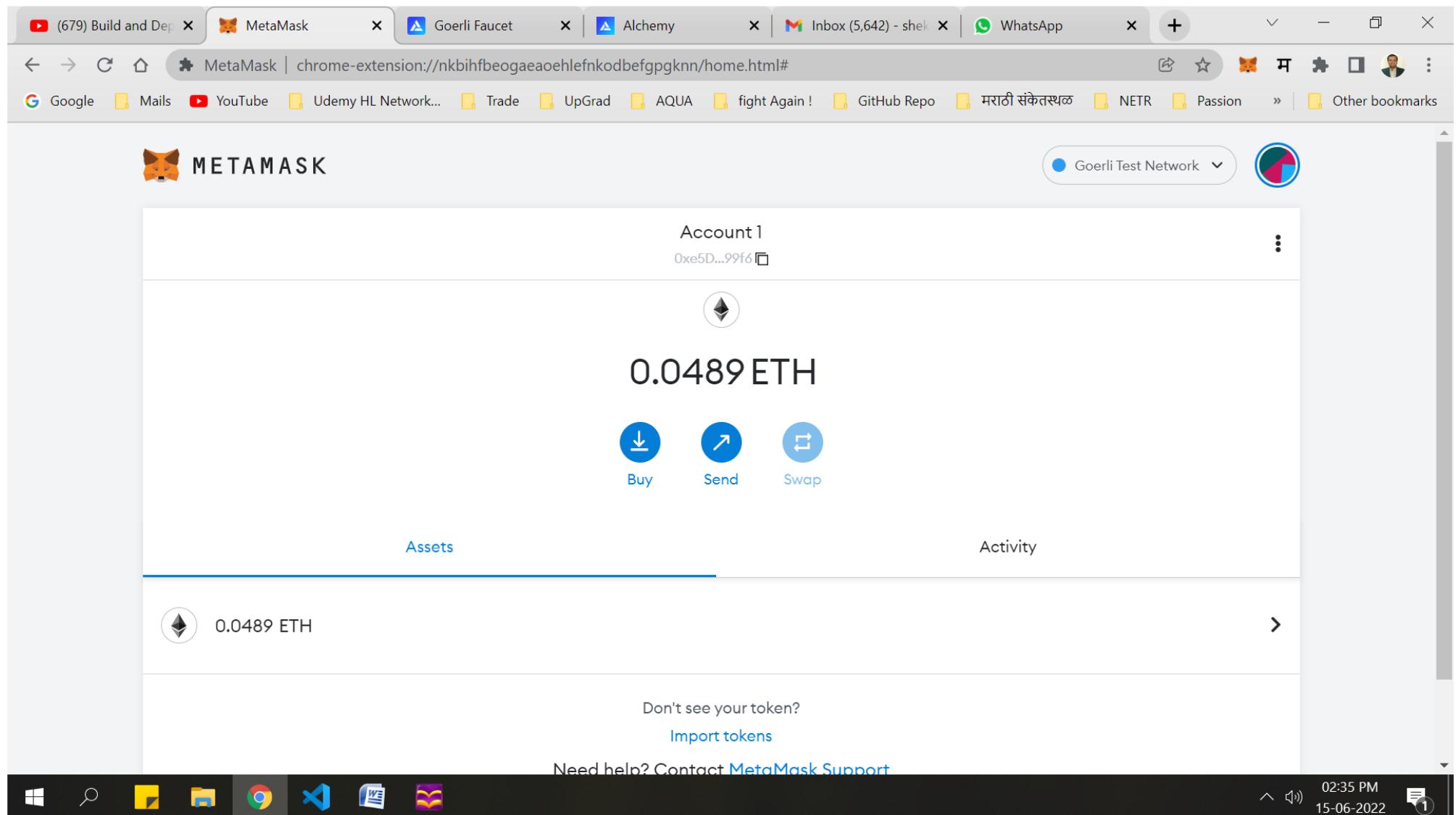
> < TERMINAL

PS C:\Users\CS\OneDrive\Desktop\web3.0\smart_contract> npx hardhat run scripts/deploy.js --network goerli
>>
Compiled 1 Solidity file successfully
Transactions address: 0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2
PS C:\Users\CS\OneDrive\Desktop\web3.0\smart_contract> []

Ln 1, Col 76 Spaces: 4 UTF-8 CRLF {} JavaScript ⚡ Go Live 🔍 02:16 PM 15-06-2022 📈 3

A screenshot of Visual Studio Code (VS Code) interface. The title bar shows "constants.js - web3.0 - Visual Studio Code". The left sidebar (Explorer) displays a project structure under "WEB3.0": client, images, node_modules, src (components, utils), and various configuration files like hardhat.config.js, Transactions.sol, Transaction.json, and deploy.js. The "utils" folder contains constants.js, which is currently selected and open in the main editor tab. The code in constants.js imports ABI from Transactions.json and exports a constant contractABI. It also defines a constant contractAddress with the value 0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2. Below the editor, the terminal shows the command "Transactions address: 0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2" and the path "PS C:\Users\CS\OneDrive\Desktop\web3.0\smart_contract>". The bottom status bar shows file statistics (Ln 7, Col 1, Spaces: 4, UTF-8, CRLF), language (JavaScript), and other details (Go Live, 02:34 PM, 15-06-2022).

```
1 import abi from './Transactions.json';
2 export const contractABI = abi.abi;
3
4
5 export const contractAddress = '0xB0314924eBf7ef1F165163476aaF8Aa19E5cfda2'
```



run Terminal Help • TransactionContext

1 Issue: 1

{} Transaction.json	TransactionContext.jsx	[vite] connecting... client.ts:16
client > src > context > TransactionContext.jsx >		[vite] connected. client.ts:53
1 import React, { useEffect, use		react-dom.development.js:29742
2 import { ethers } from "ethers		Download the React DevTools for a better development
3		experience: https://reactjs.org/link/react-devtools
4 import { contractABI, contrac		ereum
5		events: {}, _eventsCount: 1, _maxListeners:
6 export const TransactionConte		_log: u, _state: {}, ...} ⓘ
7		[[Handler]]: Object
8 const { ethereum } = window;		► deleteProperty: ()=>!0
		► [[Prototype]]: Object
		▼ [[Target]]: 1
		► chainId: "0x5"
		► enable: f()
		► isMetaMask: true
		► networkVersion: "5"
		► request: f()
		► selectedAddress: null
		► _handleUnlockStateChanged: f()

1 Issue:  1

- ▶  4 messages
 - ▶  3 user mess...
 - ▶  1 error
 - ▶  No warnings
 - ▶  3 info
 - ▶  No verbose

[vite] connecting... [client.ts:16](#)

[vite] connected. [client.ts:53](#)

[react-dom.development.js:29742](#)

Download the [React DevTools](#) for a better development experience.

experience: <https://reactjs.org/link/react-devtools>

```
> window.ethereum
```

```
< _Proxy { _events: {...}, _eventsCount: 1, _maxListeners:
```

```
100, _log: u, _state: {...}, ...} i
```

▼ [[Handler]]: Object

► **deleteProperty**: ()=>!0

► [[Prototype]]:

[Target]]: 1

chainId: "0x

enable: *f* ()

isMetaMask :

networkVersion: "

► **request: *f* ()**

selectedAddress