Centurion University
Shaping Lives ... Empowering Communities ...

School: ..... Campus: .....

Academic Year: ..... Subject Name: ..... Subject Code: .....

Semester: ..... Program: ..... Branch: ..... Specialization: .....

Date: .....

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiement :

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

React Start – DApp Frontend Scaffolding :

Step 1: Create React App :

```
# In your project root
npx create-react-app frontend
cd frontend

# Install Web3 dependencies
npm install ethers axios
npm install @web3-react/core @web3-react/injected-connector
```

Step 2: Basic React DApp Structure:

```
frontend/
├── src/
│   ├── components/
│   │   ├── Header.js          # Connect wallet button
│   │   ├── MessageList.js     # Display messages
│   │   ├── MessageForm.js     # Post new messages
│   │   └── Stats.js           # Show statistics
│   ├── hooks/
│   │   └── useBlockchain.js   # Blockchain interactions
│   ├── utils/
│   │   └── web3.js            # Web3 configuration
│   └── App.js                 # Main component
```


*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

Coding Phase: Pseudo Code / Flow Chart / Algorithm

The Evolution to React DApp:

Phase 1: Backend + Blockchain

- mart Contract on Hardhat
- Express.js Backend API
- Blockchain Integration with ethers.js
- Basic HTML frontend

Phase 2: React DApp Scaffolding

1. Wallet Integration

```
// MetaMask connection
const { account, connectWallet } = useWeb3();
```

2. Component Architecture

```
src/
├── components/          # UI Components
├── hooks/               # Custom React hooks
├── utils/               # Web3 utilities
└── App.js               # Main application
```

3. State Management

```
// React hooks for blockchain state
const { messages, stats, postMessage, likeMessage } = useBlockchain();
```

4. Real-time Blockchain Interactions

- Post messages to blockchain
- Like messages on-chain
- View real blockchain data

* Softwares used

- **Remix IDE**
- **Brave Browser**
- **Solidity**
- **Hardhat**
- **React**
- **Node.js**
- **Express.js**

**As applicable according to the experiment.*
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)

Command : npx create-react-app frontend :

```
PS D:\Talk-To-World\talk-to-world-web3> npx create-react-app frontend

Creating a new React app in D:\Talk-To-World\talk-to-world-web3\frontend.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...


added 1325 packages in 2m

271 packages are looking for funding
  run `npm fund` for details

Initialized a git repository.

Installing template dependencies using npm...

added 17 packages, and changed 1 package in 14s

 Success! Created frontend at D:\Talk-To-World\talk-to-world-web3\frontend
 Inside that directory, you can run several commands:

   npm start
     Starts the development server.

   npm run build
     Bundles the app into static files for production.

   npm test
     Starts the test runner.

   npm run eject
     Removes this tool and copies build dependencies, configuration files
     and scripts into the app directory. If you do this, you can't go back!

 We suggest that you begin by typing:

   cd frontend
   npm start
```

npm install ethers :

```
PS D:\Talk-To-World\talk-to-world-web3> cd frontend
PS D:\Talk-To-World\talk-to-world-web3\frontend> npm install ethers axios

 added 12 packages, and audited 1354 packages in 12s

 274 packages are looking for funding
   run `npm fund` for details

 9 vulnerabilities (3 moderate, 6 high)

 To address all issues (including breaking changes), run:
   npm audit fix --force

 Run `npm audit` for details.
```

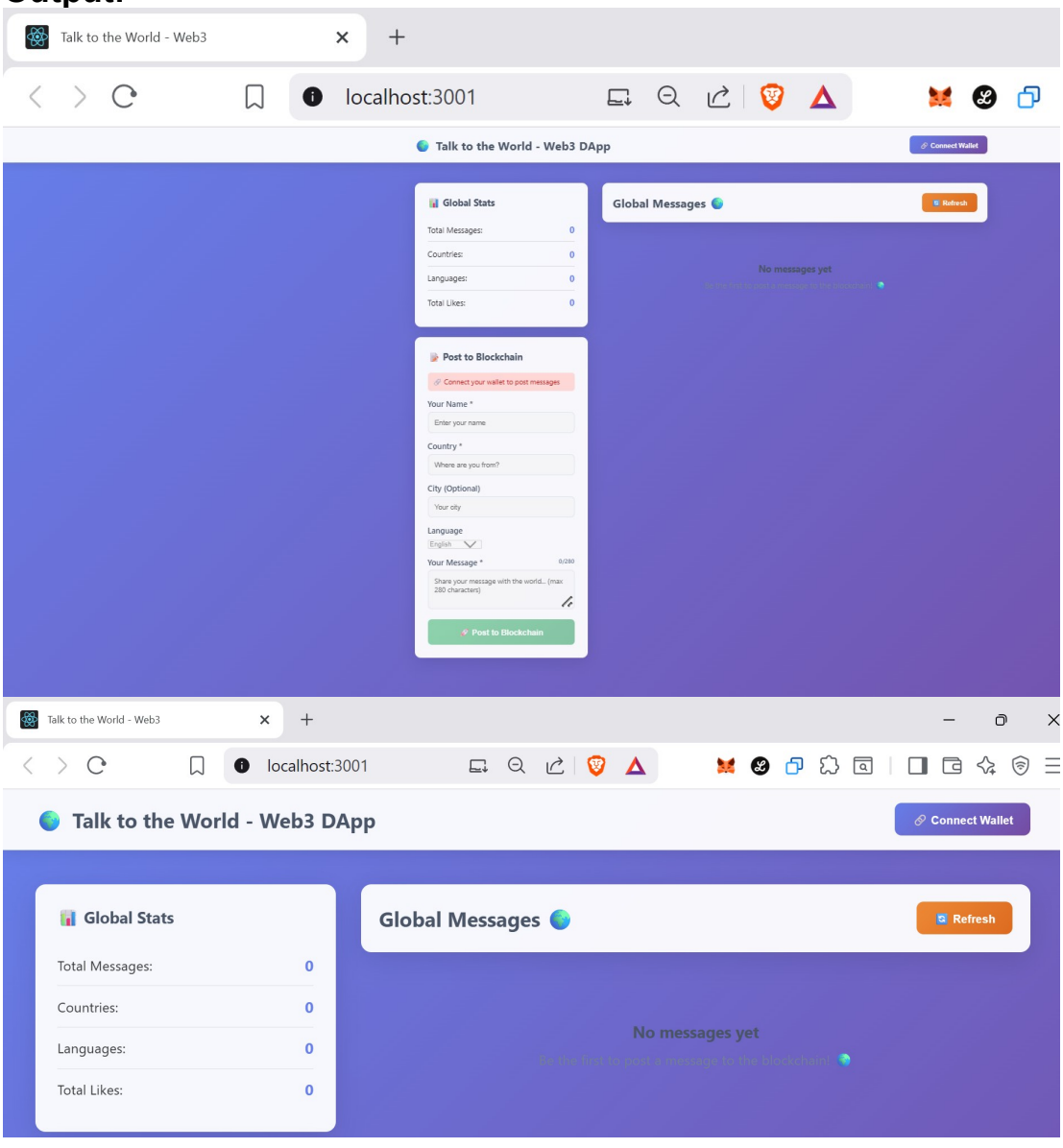

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

Implementation Phase: Final Output (no error)

npm install @web3-react/core @web3-react/injected-connector :

```
PS D:\Talk-To-World\talk-to-world-web3\frontend> npm install @web3-react/core @web3-react/injected-connector
npm warn ERESOLVE overriding peer dependency

added 41 packages, and audited 1395 packages in 23s

295 packages are looking for funding
  run `npm fund` for details

9 vulnerabilities (3 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

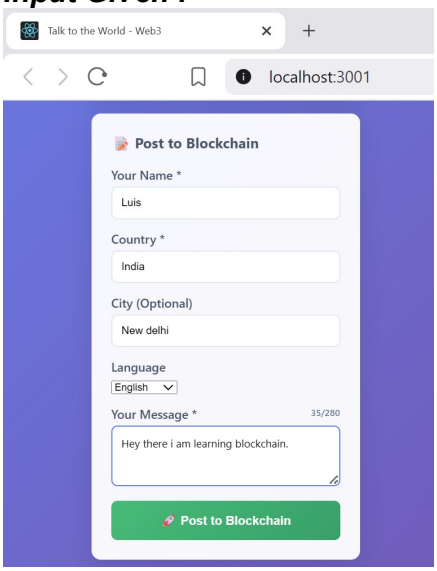
Output:



**As applicable according to the experiment.*
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)

Input Given :



* Observations

Production-ready DApp foundation that demonstrates the complete framework for a decentralized application that can now be customized and extended for any Web3 use case .

- *Web3 wallet integration*
- *Real blockchain transactions*
- *Professional UI/UX*
- *Scalable architecture*
- *Enterprise-grade code structure*

ASSESMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student:

Name :

Regn. No. :

Signature of the Faculty:

**As applicable according to the experiment. Two sheets per experiment (10-20) to be used.*