LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(An Autonomous Institution since 2010)

Accredited by NAAC with 'A' Grade & NBA (Under Tier - I),

An ISO 21001:2018, 14001:2015, 50001:2018 Certified Institution

Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada

L.B. REDDY NAGAR, MYLAVARAM, NTR DIST., A.P.-521 230.

hodcse@lbrce.ac.in, cseoffice@lbrce.ac.in, Phone: 08659-222 933, Fax: 08659-222931

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Report on one-week Industrial Project Training on Application Development in Blockchain Technologies

Event Type	: One-week industrial project training on Application Development in Blockchain Technologies
Dates	: 12/12/2024 to 18/12/2024
Time	: 09:00AM TO 04:00pm
Organized by	: Department of Computer Science and Engineering, LBRCE.
Resource Company	: TARAS System Solutions
Coordinator	: Mrs. M. Gayathri
Total no. of Students attended	: 40

Objective

To provide participants with a thorough understanding of blockchain technology, its principles, architecture, and key components. To train students in identifying and implementing blockchain-based solutions for real-world challenges across various industries. To create an engaging platform for knowledge exchange and collaboration among participants, experts, and industry professionals.

Impact Analysis

Forty final-year CSE students from the 2021 batch attended a workshop on application development using blockchain technologies. Among them, 22 students secured 16-week internships for their current semester. These initiatives have greatly contributed to the students' academic progress and career preparedness while reinforcing the institution's dedication to industry-relevant education.

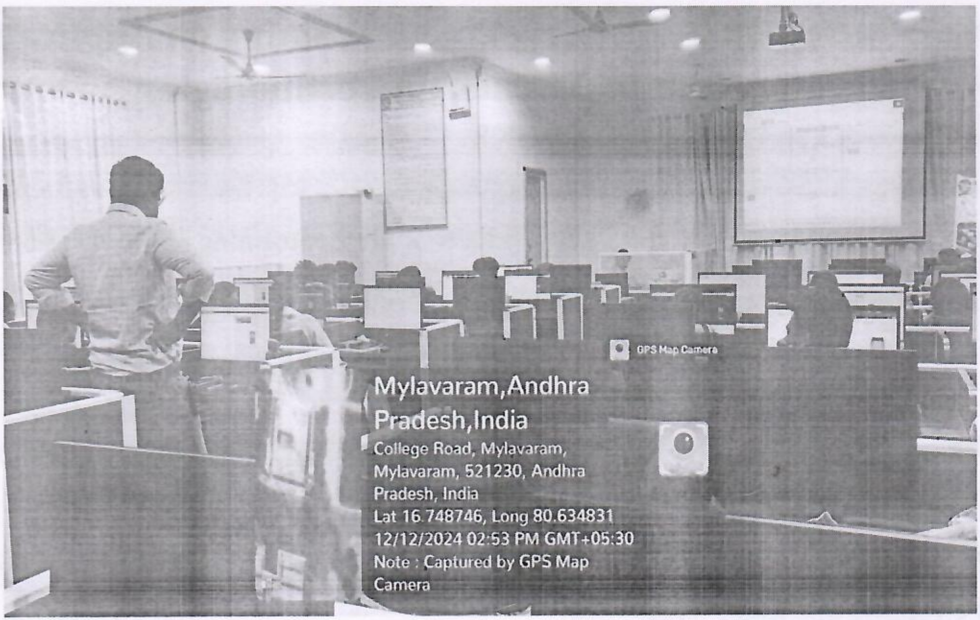
Day 1: MQTT and Introduction to Blockchain

MQTT

- Introduction to MQTT protocol.
- Key features of MQTT: lightweight, publish/subscribe model.
- Applications in IoT and real-time systems.

Blockchain

- Basics of blockchain technology.
- Importance of decentralization, immutability, and transparency.
- Key components: blocks, transactions, hashes, and ledgers.



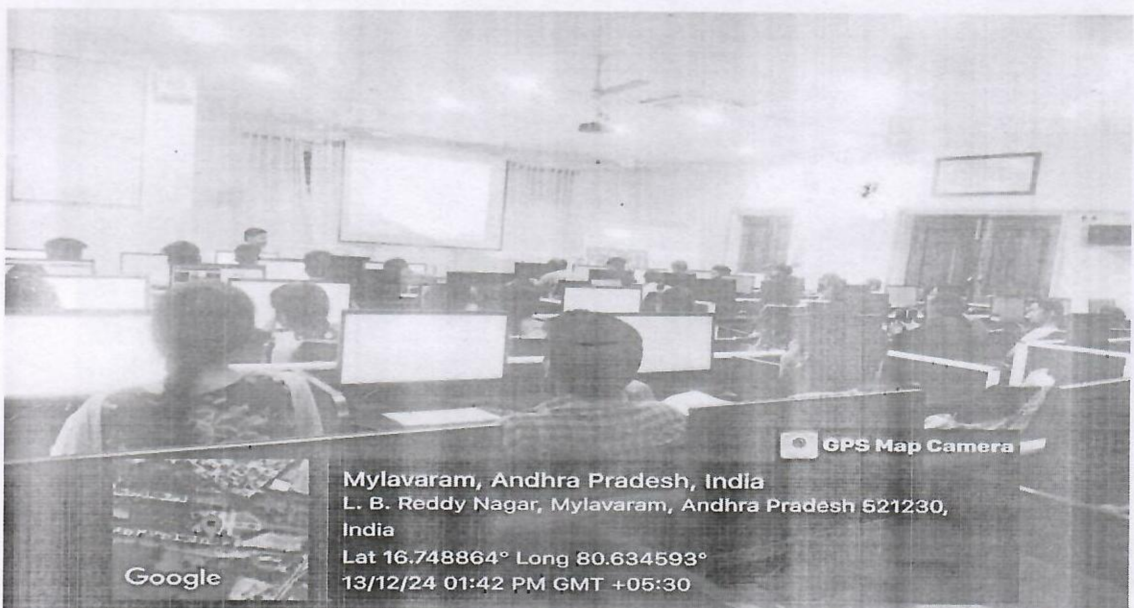
Day 2: Blockchain Creation and Core Functionalities

Blockchain Creation

- Understanding the structure of a blockchain.
- Steps to create a basic blockchain in Python.

Core Functionalities

- **Hash Validation**: Explanation of cryptographic hashing and its role in block integrity.
- **Hash Calculation**: Practical session on generating SHA-256 hashes for blocks.
- **Mining Blocks**: Implementing proof-of-work consensus algorithm.
- **Length of Blockchain**: Methods to verify the blockchain length and integrity.
- **Providing Blocks Based on Input**: Querying blocks using specific parameters (e.g., block index or transaction details).



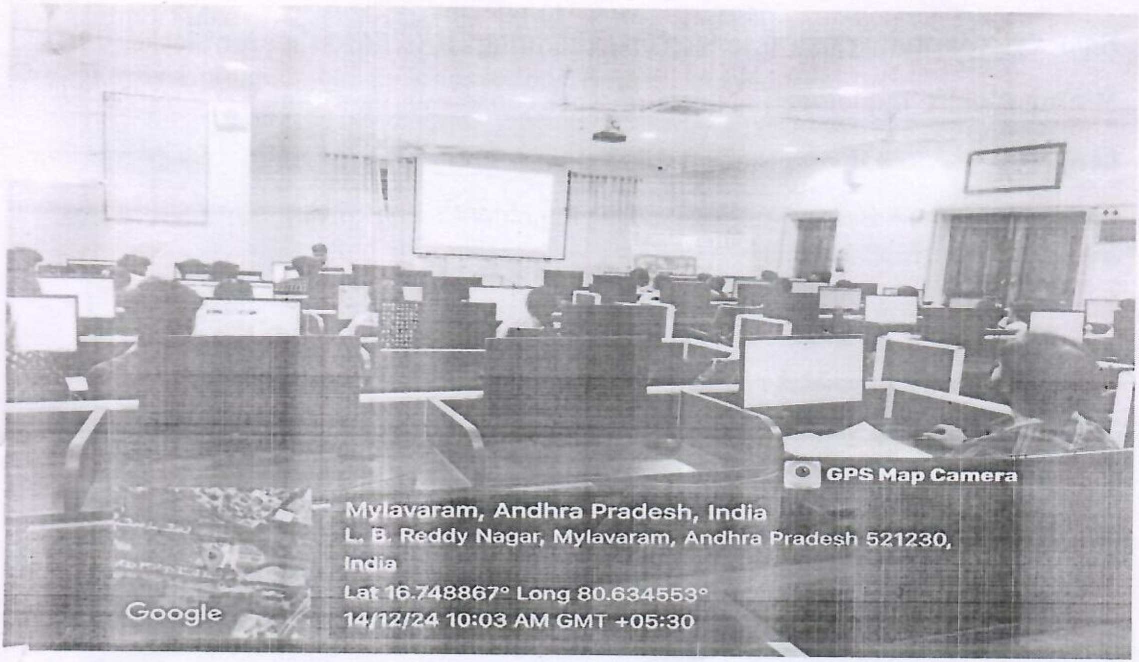
Day 3: Transactions Management

Adding Transactions

- **Manual Transaction Addition**: Demonstrated how to add transactions manually into the blockchain.
- **Input-Based Transactions**: Techniques to fetch and add transactions based on user input.

Block Formation

- Adding required transactions into a block and ensuring valid block structure.



Day 4: Advanced Consensus Mechanisms and GUI Integration

Consensus Mechanisms

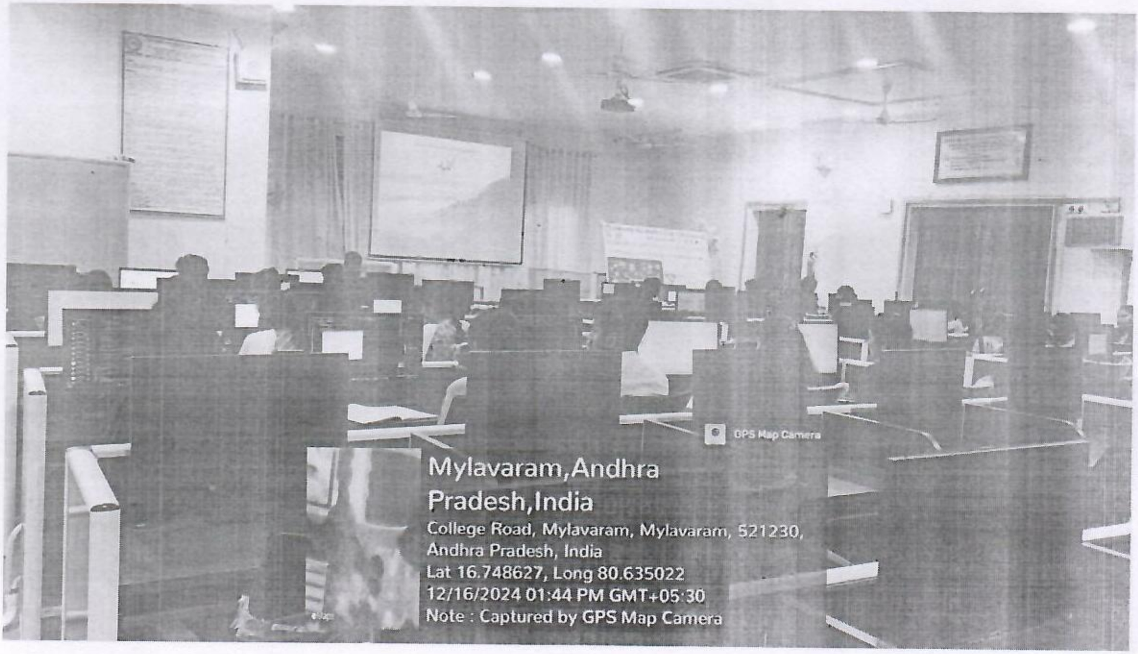
- **Proof of Burn**: Overview and implementation of proof-of-burn consensus.
- **Proof of Stake**: Detailed explanation and practical exercise on proof-of-stake.

GUI Integration

- Used Flask to create a graphical user interface for blockchain operations.
- Features implemented:
  - Adding transactions.
  - Creating and mining blocks via the GUI.

Task Distribution

- Assigned individual tasks to participants for better understanding and practice.



Day 5: Smart Contracts Development and Deployment

Smart Contracts

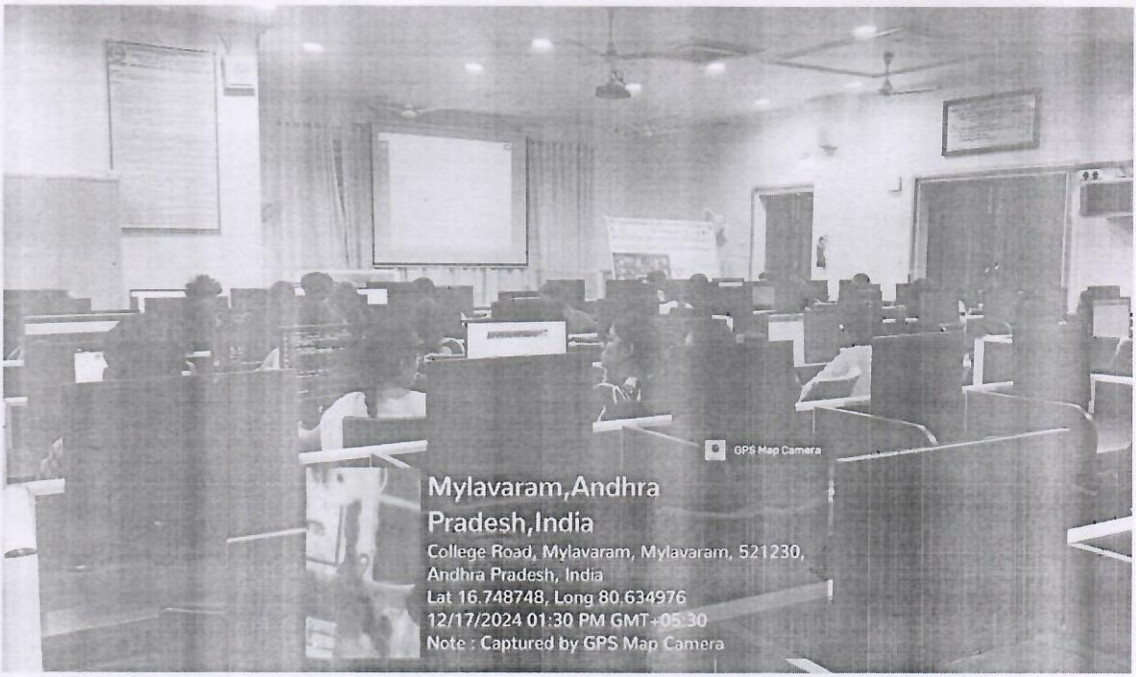
- Introduction to smart contracts and their applications.
- Hands-on session on creating basic smart contracts using Solidity.

Deployment Tools

- **Truffle**: Framework for compiling and deploying smart contracts.
- **Ganache**: Local blockchain environment for testing smart contracts.
- **MetaMask**: Wallet integration for interacting with the blockchain.

Practical Exercises

- Making transactions between accounts in Ganache.
- Demonstrated use cases of deployed contracts.



Day 6: Project Assignment and Assessment

Project Assignment

- Provided a list of ten projects for participants to select and work on.
- Participants were required to demonstrate their progress to the instructor.

Afternoon Assessment

- Conducted an assessment to evaluate understanding and practical skills.
- Focused on implementation and problem-solving abilities.

Projects presented by student

Tools and Technologies

- **Blockchain Network**: Ganache (Local Ethereum Blockchain)
- **Smart Contracts**: Solidity
- **DApp Frontend**: HTML, JavaScript (using Web3.js)
- **Wallet Integration**: MetaMask

Steps to Build the To-Do List DApp

Step 1: Setting Up the Environment

1. **Install Ganache**
   Ganache provides a lo.al Ethereum blockchain for deploying and testing smart contracts.
   - Download and install Ganache from Truffle Suite.
2. **Install Node.js**
   Install Node.js from Node.js official site.

Install Truffle
Truffle is a development framework for Ethereum smart contracts. Run the following command:

```
npm install -g truffle
```

Step 2: Create Project Structure

Create a folder for the project:
bash

```
mkdir to-do-list-dapp && cd to-do-list-dapp
```

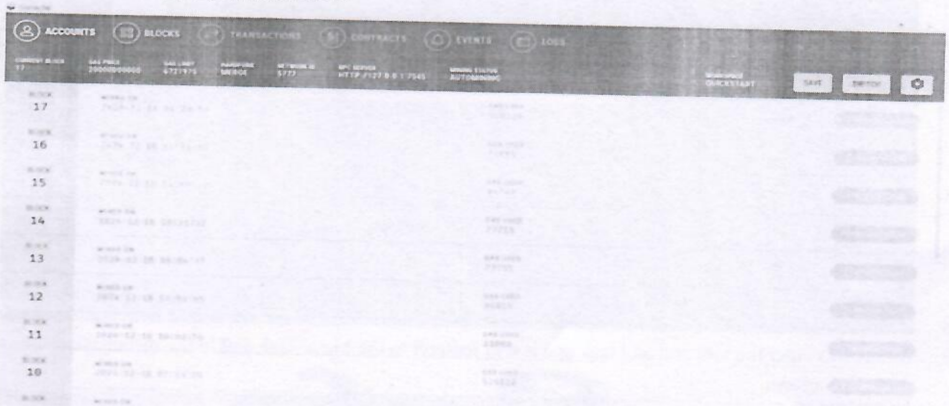
1. **Initialize the Truffle project:**
   bash
   ```
   truffle init
   ```
2. **In truffle_config.js configure networks section as in ganache**

```
development: {
    host: "127.0.0.1",     // Localhost
    port: 7545,            // Custom port
    network_id: [illegible],       // Custom network
    gas: 6721975,          // Gas sent with each transaction (default: ~6700000)
    gasPrice: 20000000000,  // 20 gwei (in wei) (default: 100 gwei)
},
```

- **The version at starting of SimpleContract.sol should match in truffle_config.js**

```
compilers: {
  solc: {
    version: "0.8.0",      // Fetch exact version from solc-bin (default: truffle's version)
      [illegible]           // Use "0.5.1" you've installed locally with docker (default: false)
      settings: {          // See the solidity docs for advice about optimization and evmVersion
       [illegible]
        [illegible]
        runs: 200
       [illegible]
      evmVersion: "byzantium"
    }
  }
},
```

Now after migrating open the ganache and check whether the block has deployed or not below screenshots from deployed blockchains



<u>Add Code to app.js</u>

<u>Next, open the app.js file and add the following code to interact with the smart contract:</u>

<u>Step 5: Connect the DApp Frontend</u>

Connect the Front-End (HTML + JavaScript)
Serves as the gateway for users to interact with the blockchain. It processes user input, sends data to smart contracts, and displays blockchain results in a visually appealing way

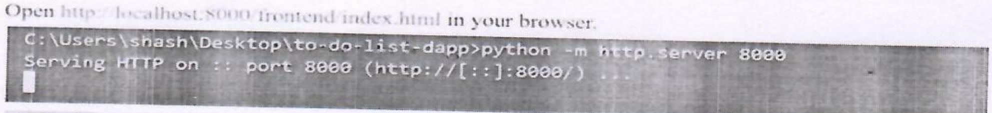
Create an index.html file in a new frontend folder:

<u>Step 6: Run the Application</u>

Start a simple HTTP server:

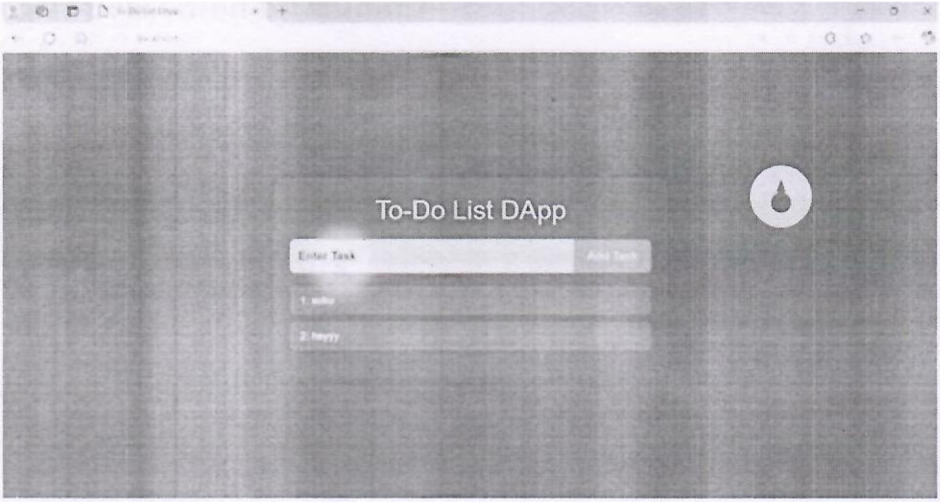
python -m http.server 8000

1. Open http://localhost:8000/frontend/index.html in your browser.



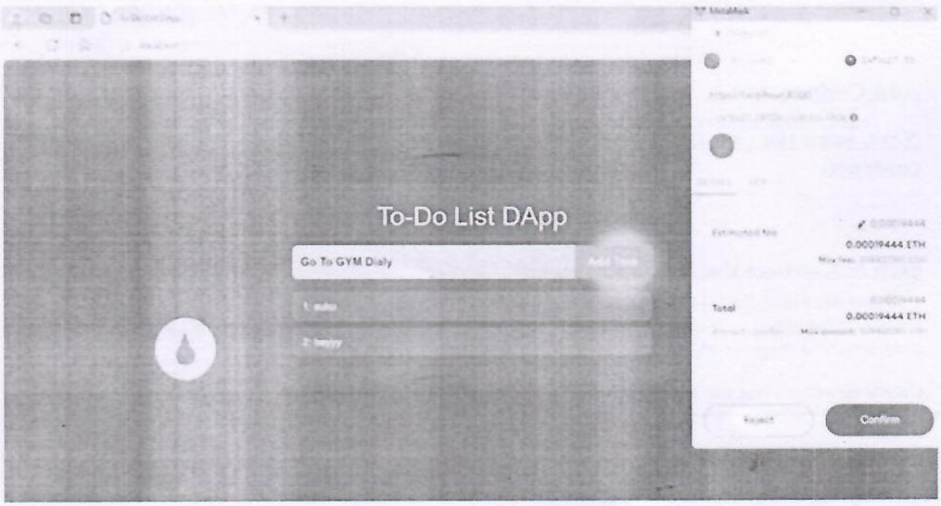
2.

3.
4. The brower and after opening it it will popup a MetaMask
5. And after entering the password of the MetaMask



6.

7. We need to enter the task and add task and it will redirect to the MetaMask and in the metamask click on conform.



8.

OUTCOME

- Students developed a comprehensive understanding of blockchain concepts, including its architecture, functionalities, and various practical applications across industries.
- Through guided exercises, they gained practical skills in creating blockchain networks, understanding consensus mechanisms, managing transactions, and deploying smart contracts.
- Students applied their learning in final projects, demonstrating their ability to integrate blockchain solutions in real-world scenarios.
- The workshop's structured assessments and interactive sessions boosted students ability to analyse blockchain use cases and propose innovative solutions.
- Many attendees received certificates recognizing their expertise, enhancing their career prospects in blockchain-related fields.

- The collaborative environment fostered connections among participants, instructors, and industry experts, paving the way for future collaborations.

Overall, the workshop empowered participants with technical expertise and practical insights, preparing them to contribute effectively to blockchain innovation and adoption.

NEWS Paper cuts

సాక్షి

ఇంటర్న్‌షిప్‌పై శిక్షణ

మైలవరం: మైలవరం లకిరెడ్డి బాలిరెడ్డి ఇంజినీరింగ్ కళాశాలలో కంప్యూటర్ సైన్స్ విభాగం ఆధ్వర్యంలో ఇంటర్న్‌షిప్‌పై శిక్షణ కార్యక్రమం గురువారం ప్రారంభమైంది. కంప్యూటర్ సైన్స్ విభాగాధిపతి డాక్టర్ డి. వీరయ్య మాట్లాడుతూ ఇంటర్న్‌షిప్ అనేది విద్యార్థులకు విద్య. కెరీర్ మధ్య ఉన్న గ్యాప్‌ను తీర్చే ప్రధాన మార్గమన్నారు. టూరాస్ సాఫ్ట్‌వేర్ సొల్యూషన్స్ సంస్థ ఆధ్వర్యంలో జరిగిన ఈ కార్యక్రమంలో తృతీయ సంవత్సరం సీఎస్ఈ విభాగాల్లో బ్లాక్ చైన్ టెక్నాలజీ. ఏఐ అండ్ ఎంఎల్. ఫుల్ స్టాక్ నాలుగో సంవత్సరం విద్యార్థులు పాల్గొన్నారు.

13/12/2024 | NTR(Tiruvuru) | Page : 9
Source : https://epaper.sakshi.com/

లకిరెడ్డి బాలిరెడ్డి కాలేజ్ ఆఫ్ ఇంజనీరింగ్ (LBRCE), మైలవరంలో కంప్యూటర్ సైన్స్ విభాగము లోడాక్టర్ డి.వీరయ్య పర్యవేక్షణలో టారాస్ సాఫ్ట్‌వేర్ సొల్యూషన్స్ వారు 3 సి.ఎస్.ఇ (CSE) విభాగాల్లో (బ్లాక్‌చైన్ టెక్నాలజీ, ఏఐ & ఎంఎల్ మరియు ఫుల్ స్టాక్) 4వ సంవత్సర విద్యార్థులకు ఇంటర్న్షిప్ ట్రైనింగ్ 12/12/24 నుంచి 18/12/24 వరకు ఈ ప్రోగ్రాం నిర్వహిస్తున్నారు. ఇంటర్న్షిప్ అనేది విద్యార్థులకు విద్య మరియు కెరీర్ మధ్య ఉన్న గ్యాప్‌ను తీర్చే ప్రధాన మార్గం. విద్యార్థులను ప్లేస్‌మెంట్స్ కోసం మరింత అర్హులుగా నిలబెడతాయి అని కంప్యూటర్ సైన్స్ విభాగాధిపతి డాక్టర్ డి. వీరయ్య తెలిపారు.

#AndhraPradesh

Conclusion

The training provided a comprehensive understanding of blockchain concepts and practical applications. Participants gained hands-on experience in blockchain creation, consensus mechanisms, transaction management, and smart contract deployment. The final projects and assessments helped solidify their learning and application of knowledge.

Mrs M Gayathri

COORDINATOR

Dr. D. Veeraiah

(Head of the Department)