# BLOCKCHAIN PROJECT

# CampusCoins - Campus Token Rewards System

## Synopsis:

A Blockchain-based token reward system that allows students to earn and redeem tokens using the barcode on their college ID cards. The system will promote and enhance student engagement in extracurricular and academic activities and provide incentives by allowing students to redeem tokens for discounts and offers at the campus canteen, bookstore and other facilities.

Example:

- A Student scores above 35 in ISA's , then he will earn 100 tokens.
- Student attends Hackathon: Earns 50 tokens
- Student attends a guest lecture , he earns 20 tokens etc…

The project involves creating a system where:

1. **Token Generation and Storage:** Tokens are generated and stored on a blockchain (like Polygon) for transparency and security.
2. **Web Interface:** Event head uses a barcode scanner to scan the barcode on participants college ID cards to:
    - Credit tokens for attending events or participating in campus activities.
    - The Student can check credited tokens on the website.
    - Student can redeem tokens at the campus canteen or bookstore for discounts or free items.
    - The Student can check the balance after deduction of tokens from the website.
3. **Token Transactions:** Token credits and debits are processed through smart contracts on the blockchain.
4. **Bookstore/Canteen Side:** Canteen staff scan the barcode, which verifies the token balance and processes the redemption.

## Functional Requirements:

### 🎯 Student Side:

Students should be able to:

- Scan barcode to earn tokens.
- View token balance in real-time.
- Redeem tokens at the canteen.
- Receive notifications on successful transactions.

### 🎯 Event Organizer Side:

Event organizers should be able to:

- Set token rewards for specific events.
- Generate and manage event IDs.

### 🎯 Canteen Side:

Canteen staff should be able to:

- Scan barcodes to redeem tokens.
- View the token redemption status.
- Approve or reject token-based transactions.

### 🎯 Admin Side:

Admin should be able to:

- Manage token supply.
- View and approve events.
- Monitor transaction history.

## System Requirements:

### 🎯 FrontEnd Requirements:

Framework: React.js

### 🎯 Backend Requirements:

Framework: Node.js + Express.js

Database Requirements: MongoDB (for storing metadata like user info, event details transactions will be saved on blockchain)

Blockchain Interaction: Web3.js

#### 🎯 Blockchain Requirements:

Blockchain: Polygon Testnet

Smart Contract Language: Solidity

Wallet Integration: MetaMask