Blockchain Development on Celo

SHIELD
Filling the entrepreneur's missing pieces

Instructor: Allan

Course Objectives

1. Introduce students to blockchain basics, focusing on decentralized technologies and real-world applications.

2. Provide practical skills in smart contract development with Solidity and dApp creation on Celo.

3. Contextualize blockchain's potential for financial inclusion in Kenya and beyond.

Week 1: Introduction to Blockchain Technology

1. **Objective**: Understand blockchain fundamentals, including concepts like decentralization and consensus mechanisms.

2. **Key Terms**: Blockchain, Decentralization, Distributed Ledger, Consensus, Proof of Work (PoW), Proof of Stake (PoS).

Week 1: Introduction to Blockchain Technology

What is Blockchain?

- We start with the concept of a “ledger” and discuss how a blockchain is a special kind of distributed ledger that records transactions across many computers.

Core Concepts:

- **Decentralization**: No central authority. We’ll compare with Kenya's mobile money network, which, although centralized, enables broad access and builds trust.
- **Distributed Ledger**: Every participant (node) has a copy, making it transparent and secure.

Week 1: Introduction to Blockchain Technology

Types of Blockchains:

- **Public** (like Bitcoin) vs. **Private** (like Hyperledger).
- Examples: Bitcoin's public ledger vs. a private network for businesses.

Consensus Mechanisms:

- Describe PoW and PoS, using simplified examples (like the idea of voting or mutual trust).

Week 1: Assignment

Assignment:

- Write a short summary on how blockchain works, including a real-world use case like cryptocurrencies or decentralized finance (DeFi).

- **Deliverable**: A one-page summary.

Week 2: Introduction to Celo and Financial Inclusion

Objective: Explore Celo's mission to make financial tools accessible, especially for underserved populations.

Key Terms: Celo, Stablecoin, cUSD, Validator, Financial Inclusion.

Week 2: Introduction to Celo and Financial Inclusion

Celo Overview:

- Introduction to Celo's goals: bringing mobile-friendly blockchain access to people without traditional bank access.

Celo Architecture:

- PoS-based, making it energy-efficient.
- The role of Validators (similar to bank representatives, but decentralized).

Week 2: Introduction to Celo and Financial Inclusion

Stablecoins:

- Explain cUSD and cEUR as digital currencies pegged to fiat currencies (USD and EUR) to avoid volatility.
- Example: How stablecoins can be more predictable for day-to-day transactions than cryptocurrencies.

Celo vs. Ethereum and Binance Smart Chain:

- Differences between Celo, which is mobile-first and focuses on low-cost transactions, and other platforms

Week 2: Assignment

Assignment:

- Research how Celo helps promote financial inclusion in Kenya or East Africa, and write a brief summary of one impactful project.

- **Deliverable**: 1-2 page research summary.

Week 3: Solidity Basics and Smart Contracts

Objective: Learn Solidity and build simple smart contracts.

Key Terms: Solidity, Smart Contracts, State Variables, Functions, Remix IDE.

Week 3: Solidity Basics and Smart Contracts

Introduction to Solidity:

- Solidity is the language used to write smart contracts (programs that execute automatically on the blockchain).

- Demo: Write a simple “Hello Kenya” smart contract using Remix

Week 3: Solidity Basics and Smart Contracts

Data Types and Control Structures:

- Basic data types like integers, booleans, and strings.

- Control flow: if-statements, loops, and functions

Week 3: Solidity Basics and Smart Contracts

Using Remix IDE:

- Hands-on session where students set up and explore Remix, an online tool for Solidity development.

- We'll write a simple contract to store and retrieve data, e.g., a student's name or favorite food.

Week 3: Assignment

Assignment:

- Create a basic contract on Remix that allows users to set and retrieve a message.

- **Deliverable**: Contract code and a reflection on the experience.

Week 4: Setting Up a Development Environment for Celo

Introduction to Development Tools

- **Key Topics**:
  - Overview of essential tools: Node.js, npm, Git, Visual Studio Code, Celo CLI, ContractKit.

- **Discussion Prompt**:
  - “Why is each of these tools necessary for blockchain development?”

Week 4: Setting Up a Development Environment for Celo

Step-by-Step Installation

- **Topics Covered**:
  - Detailed installation guide for Node.js, npm, Git, and VS Code.
  - Setup guide for Celo CLI and Viem, connecting to the Alfajores Testnet.
- **Hands-On Exercise**:
  - Students follow along to install each tool.
  - Troubleshooting common setup issues.
- **Checkpoints**:
  - Pause after each step for questions and troubleshooting.

Week 4: Setting Up a Development Environment for Celo

Testing the Setup

- **Exercise**:
  - Students confirm setup by checking if they can interact with the Celo network.

- **Assignment**:
  - Write a setup guide detailing their process and how they resolved any setup issues.

Week 5: Capstone Project - Building Your Own dApp on Celo

- **Lesson Objective**: Apply all learned skills to build a fully functional dApp on Celo.

Week 5: Capstone Project - Building Your Own dApp on Celo

Planning and Designing Your dApp

- **Exercise**:
  - Break students into small groups to brainstorm dApp ideas relevant to Kenya (e.g., micro-lending app, digital wallet, local marketplace).
- **Project Requirements**:
  - Define core functionality, the purpose of the dApp, and design considerations.
- **Illustrative Example**:
  - Show a sample dApp design with features, flow, and user interface suggestions.

Week 5: Developing Smart Contracts and Testing on Alfajores

- **Key Topics**:
  - Writing, deploying, and testing contracts on the Alfajores Testnet.
- **Exercise**:
  - Students deploy their first contract, perform a token transfer, and test user authentication.
- **Discussion Prompt**:
  - “What challenges did you face, and how did you resolve them?”

Week 5: Finalizing the dApp and Documentation

- **Project Requirements**:
  - Write a report covering dApp's purpose, technical processes, and challenges.
- **Final Presentation**:
  - Each student presents their dApp, shares code snippets, and discusses technical challenges and solutions.
- **Assessment**:
  - Peers and instructor provide feedback, highlighting innovative solutions and suggesting improvements.

THANK YOU

SHIELD Learn