

BuckeyeCoin - Second Deliverable - DEPLOYMENT

Deployment & Transactions | Team Roles Highlighted

Overview

This deliverable summarizes the *deployment* of BuckeyeCoin on the testnet, provides transaction evidence, and distinctly outlines what each team (CS and Finance) contributed to make this step successful.

1. Computer Science Team Responsibilities

A. Contract Deployment

- Finalized smart contract code for BuckeyeCoin (token parameters from tokenomics doc)
- Chose testnet (e.g., Ethereum Goerli, Solana Devnet)
- Deployed contract to the network
- Produced proof:
 - Contract address: [Insert address here]
 - Deployment logs and screenshots (blockchain explorer)

B. Minting & Test Transactions

- Minted initial supply to team wallets
- Executed minimum one transfer between two accounts
- Captured results:
 - Transaction hashes, explorer screenshots
 - Verified tokens visible in wallets

C. Wallet Integration

- Prepared guide:
 - How to set up [MetaMask/Phantom] for the chosen network
 - How to add custom BuckeyeCoin token
 - How to send and receive tokens

2. Finance Team Responsibilities

A. Transaction Analytics & Logging

- Monitored real-time blockchain activity with explorer
- Logged mint and transfer transactions, including:
 - Date, hash, recipient, fee, and action type
- Maintained spreadsheet with ongoing analytics

B. Value & Use Case Analysis

- Drafted initial “campus use” value peg for BuckeyeCoin (e.g., 1 BuckeyeCoin = 1 club vote or event admission)
- Evaluated implications of supply, transfer fees, and user accessibility

C. Economic Commentary

- Discussed incentive design, initial stability plan (note: basic at this stage)
- Reflected on lessons learned and next steps for broader campus adoption

3. Joint Deliverable Summary

Proof of Success:

- Live contract address and screenshots, with transaction logs
- Table of actions (see example below):

Date	Tx Hash	Action	Fee	Recipient	Logged by (Team)
11/12	0xabcd...123	Mint	\$0.01	Alice	CS
11/12	0xdef...456	Transfer	\$0.005	Bob	Finance

Next Steps:

- CS preps campus use-case integration, interfaces
- Finance refines usage incentives, analytics, governance proposals

4. Visuals & Artifacts

- screenshots (explorer, wallet, logs)
- Excerpts from analytics sheets

Collaboration Reflection

“Cross-team collaboration let CS deploy the token and teach Finance how to use wallets and analyze data. Finance’s feedback on use-case, transaction tracking, and incentives shaped our deployment priorities. Both teams are now ready to attack campus use-case integration.”

Search Prompts

- “How to deploy SPL token on Solana Devnet with Solana CLI”
- “Step-by-step tutorial: creating and minting Solana tokens for college projects”
- “Integrate Phantom wallet with custom SPL token on Solana Devnet”
- “Remix IDE deploy ERC-20 token contract on Ethereum Goerli testnet”
- “Compare Solana and Ethereum contract deployment workflows”

GPT Prompts

- “Explain the process for deploying an SPL token on Solana Devnet using Solana CLI, with troubleshooting advice for common errors.”
- “Draft an ERC-20 token contract in Solidity for Remix IDE, ready for deployment on Goerli testnet.”
- “Create a wallet guide for students to add a Solana SPL token to Phantom wallet and an ERC-20 token to MetaMask.”
- “Design an analytics spreadsheet for tracking Solana Devnet token transactions, including address, date, tx hash, and fee.”
- “What are best practices for verifying Solana contract address and token supply post-deployment?”

YouTube Video Search Prompts

- “Deploy SPL token on Solana Devnet step-by-step tutorial”
- “Phantom wallet: adding and sending SPL tokens demo”
- “Remix IDE: deploy ERC-20 contract on testnet (Goerli)”
- “Solscan explorer: tracking Solana token transactions”
- “Solana CLI quickstart: minting and transferring SPL tokens”