Blockchain Fundamentals, Sub Code: **7KS05**

TEC (Teaching Evaluation Component)

On

Title

BlockFund: Decentralized Blockchain Crowdfunding Management System

Submitted By

(03) Devashree. U. Pundlik

(65) Sarthak.M.Deshmukh

(66) Shantanu. R. Dongare

(67) Shivam.A.Gahale

Final year, Seventh Semester

B.E. (Computer Science and Engineering)

Department of Computer Science & Engineering Shri Sant Gajanan Maharaj College of Engineering, Shegaon – 444 203

2025-2026

1. Introduction

BlockFund is a decentralized crowdfunding platform that replaces opaque, intermediary-driven fundraising with a transparent, blockchain-based model where every contribution is immutably recorded on a public ledger. Traditional platforms democratized access to capital but still struggle with fraud, limited transparency, high fees, delays, and weak accountability. Blockchain overcomes these gaps through peer-to-peer transactions, cryptographic hashing, and chained blocks that prevent tampering, while smart contracts can automate milestone-based fund release and refunds to strengthen backer protection. BlockFund implements SHA-256 for data integrity, a proof-of-work mechanism for validating blocks, and a Flask-powered web interface to make secure, auditable donations simple for creators and backers. The result is a secure, efficient, and verifiable crowdfunding environment that eliminates reliance on centralized third parties and significantly improves trust, accountability, cost, and user experience.

2. Importance

1. Addressing Trust and Transparency Gaps
2. Enhanced Security and Fraud Prevention
3. Cost Reduction and Financial Efficiency
4. Democratization of Investment Access
5. Immutable Record-Keeping and Auditability
6. Community Building and Gamification
7. Solving Real-World Fundraising Challenges
8. Alignment with Web3 and Decentralization Trends

3. HARDWARE SOFTWARE REQURIED

Hardware

1. Computer (64-bit CPU, 8–16 GB RAM, SSD storage)
2. Reliable Internet Connection

Software

1. Python 3.10+ with pip/venv
2. Flask and Werkzeug
3. Database: SQLite
4. Web Browser (Chrome/Firefox/Edge)

4. System Description and Architecture

System Description

BlockFund is a web-based decentralized crowdfunding platform that combines a conventional web application stack with a built-in blockchain ledger to provide end-to-end transparency for pledges and fund flows. The application exposes RESTful endpoints via a Flask backend, persists user and campaign metadata in a relational store (SQLite for the demo), and records donations as transactions that are mined into blocks secured by SHA-256 hashing and a lightweight proof-of-work routine. The frontend (HTML/CSS/JavaScript) renders campaign listings, creator profiles, donation flows, and a blockchain explorer, emphasizing clarity for non-technical users through tooltips, progress indicators, and validation prompts. Each donation creates a transaction object appended to a pending pool; mining seals the pool into a block that references the previous block's hash, producing an immutable, auditable chain. Wallet balances and campaign totals are synchronized with on-chain transaction hashes to ensure that what users see in the UI corresponds to verifiable ledger entries. This hybrid approach gives the familiarity and speed of a classic web app while delivering the auditability and tamper resistance of a blockchain.

Architecture Overview

The system is organized into three cooperating layers:

- **Presentation Layer (Client/UI)**
    - Technologies: HTML5, CSS3, JavaScript.
    - Responsibilities: Authentication screens, campaign creation and browsing, donation forms, progress bars, notifications, tooltips, public profiles, leaderboards, and a blockchain explorer view that lists blocks, proofs, and transaction details.
    - Interactions: Consumes JSON endpoints over HTTPS; renders server-side templates via Jinja and hydrates dynamic areas with fetch-based requests.
- **Application Layer (Flask Backend)**
    - Technologies: Python 3, Flask 3.x, Werkzeug.
    - Security: Password hashing, session cookies, server-side checks for wallet balance and input ranges; environment-based secret keys.
- **Data and Ledger Layer**

- Relational Store (SQLite): Users, campaigns, donations metadata, and derived aggregates for fast UI queries. Designed to be swappable with PostgreSQL in multi-user deployments.

Blockchain Module: In-memory (and serializable) chain with:

- Block structure: index, timestamp, transactions[], proof, previous_hash.
- Cryptographic integrity: SHA-256 over a canonical JSON representation.
- Consensus: Simple proof-of-work (e.g., four leading zeros) to control block creation rate and provide verifiable work.
- Chain validation: Checks previous_hash linkage and valid_proof across the entire ledger.

5. Data Schema (On-Chain Storage)

BlockFund maintains an application-level blockchain where each donation is encoded as a transaction and permanently committed within mined blocks. The ledger is append-only and validated via SHA-256 hashing and a lightweight proof-of-work. All fields are serialized to canonical JSON before hashing to ensure deterministic digests.

Block Structure

- index: Integer — Sequential position of the block in the chain
- timestamp: Float/Unix seconds — Block creation time at commit.
- transactions: Array<Transaction> — Ordered list of pending transactions sealed in this block.
- proof: Integer — Nonce discovered by the proof-of-work algorithm satisfying the difficulty predicate.
- previous_hash: String(64 hex) — SHA-256 hash of the prior block's canonical JSON.

Genesis Block

- previous_hash: "1" (constant), proof: 100 (constant), transactions
- Established at node initialization to anchor the chain.

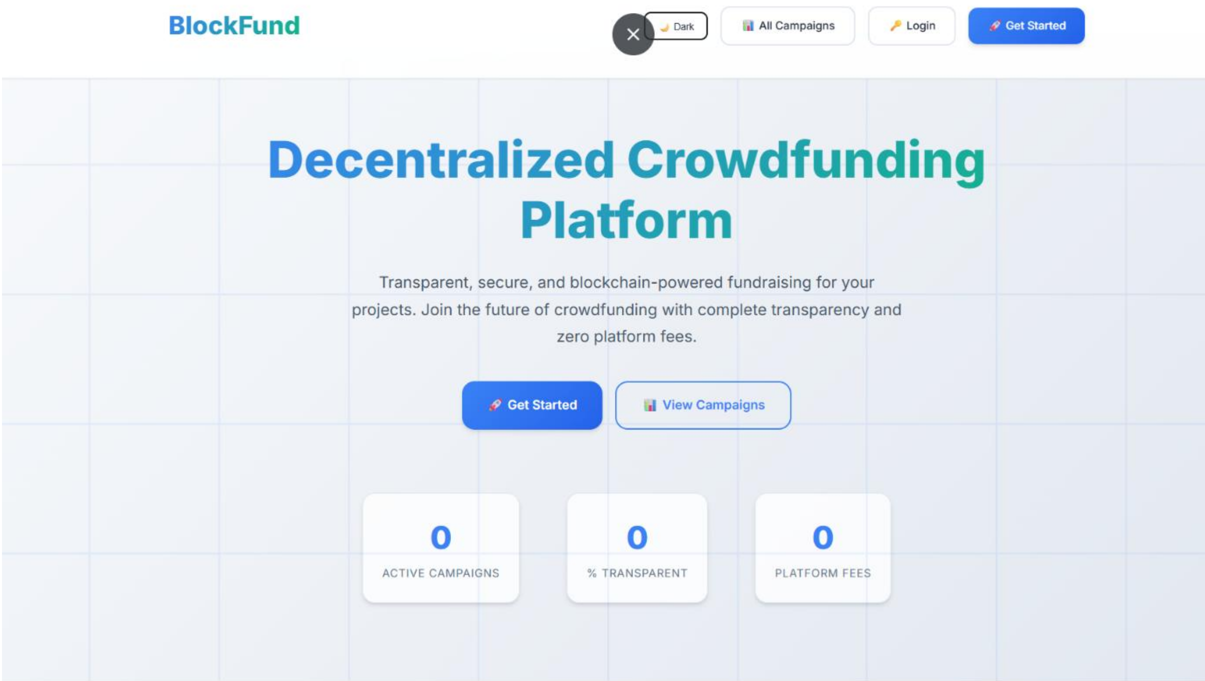
Difficulty and Valid Proof Predicate

- Difficulty: 4 leading zeros for demo environments (configurable).

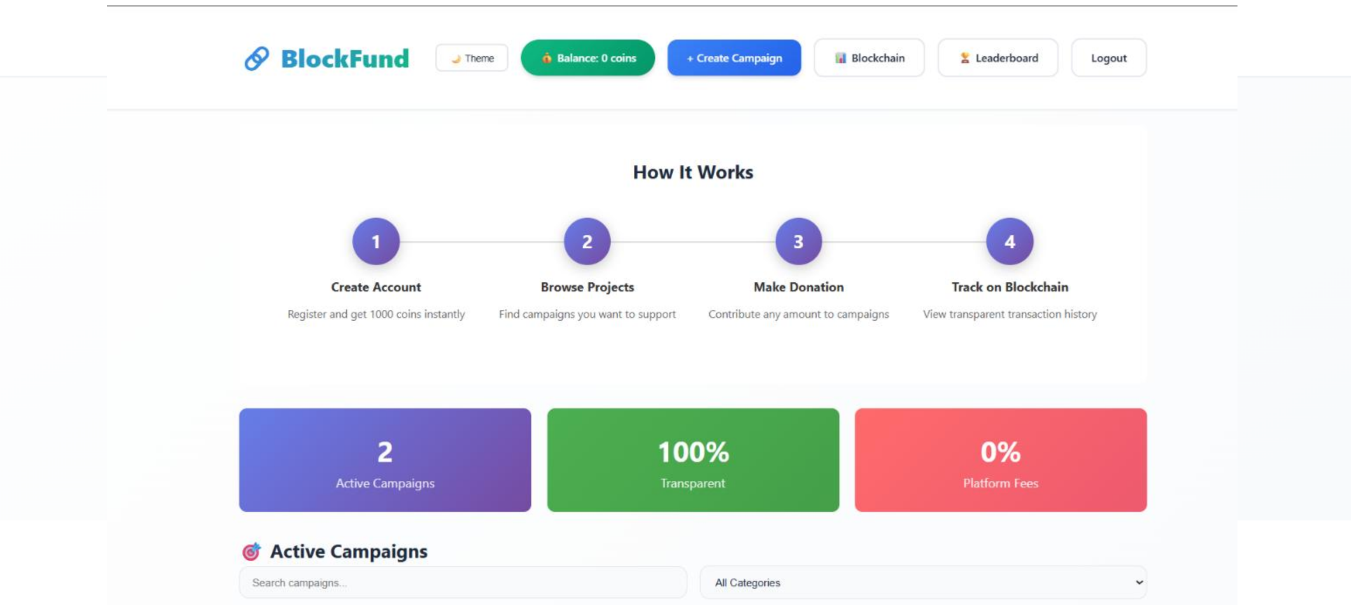
6 . GUI / Frontend Appearance

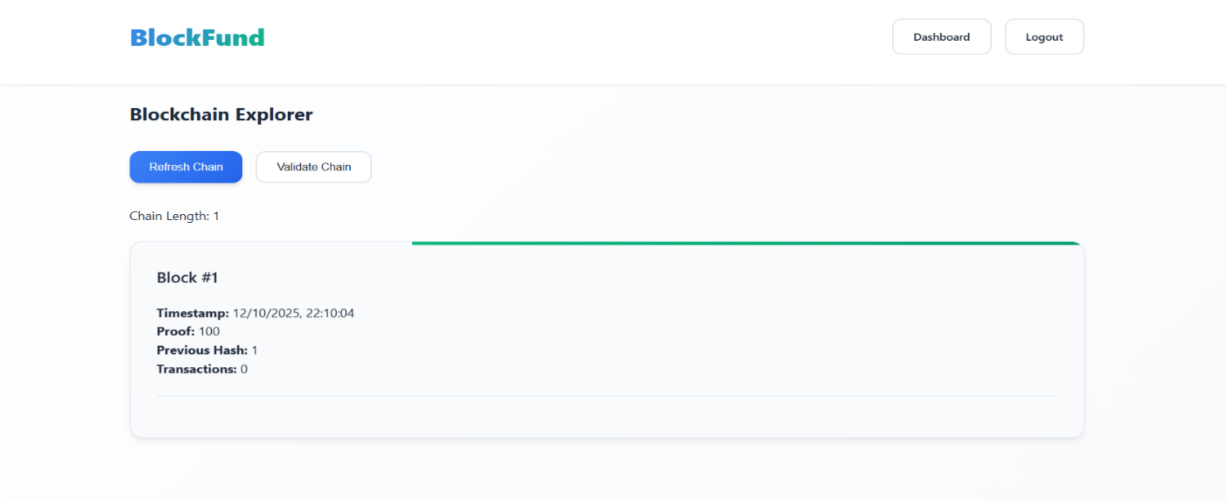
The user interface is designed to be clean and intuitive, allowing for easy interaction for the user

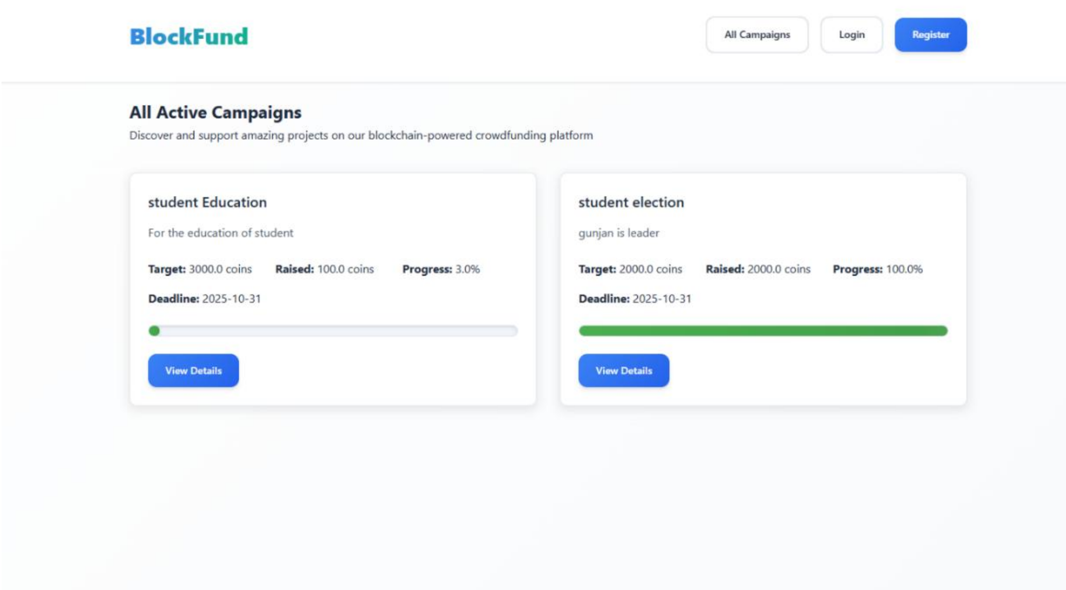
Main Dashboard

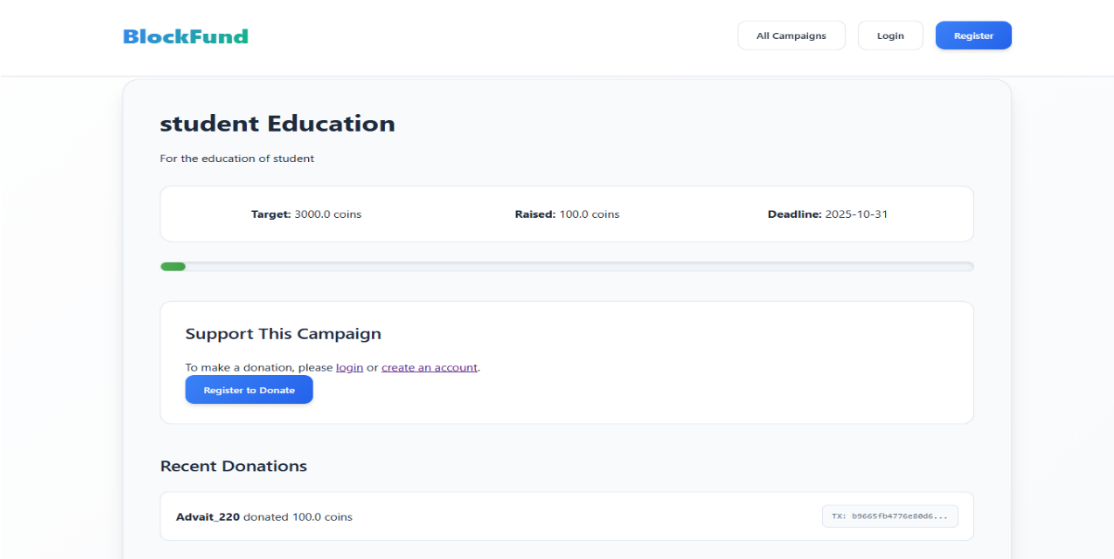


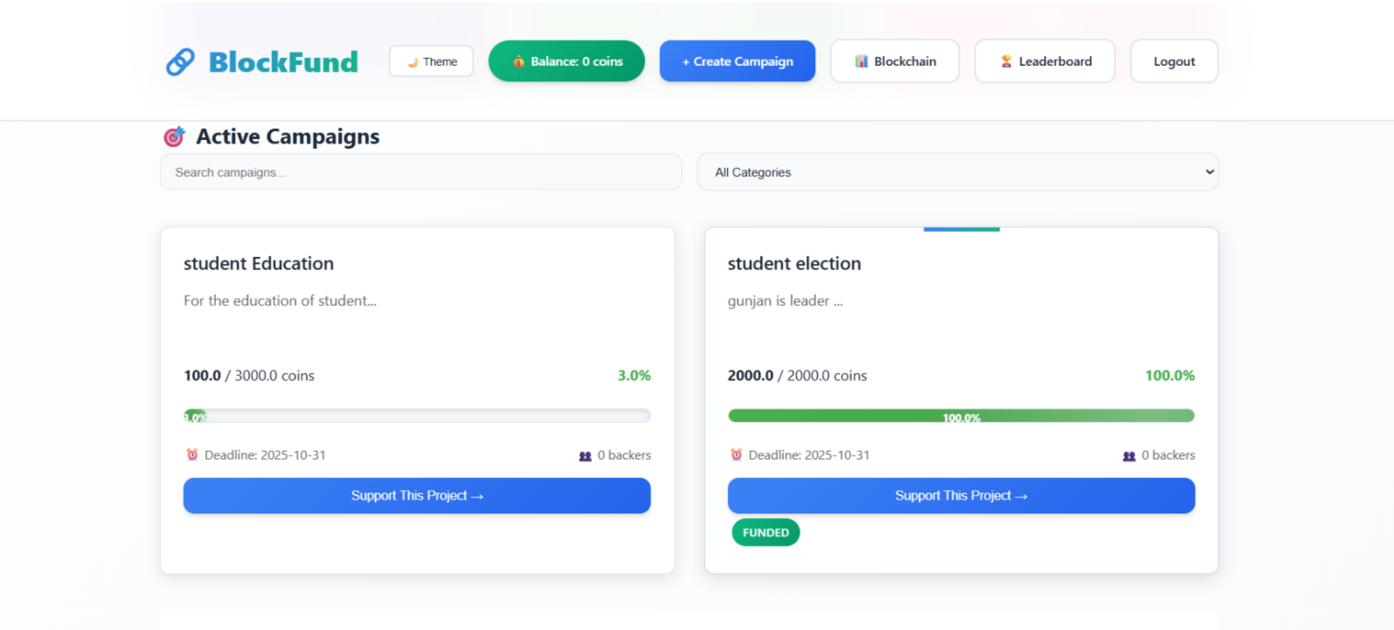
Screenshots of project




BlockFund
Dashboard
Logout
Blockchain Explorer
Refresh Chain
Validate Chain
Chain Length: 1
Block #1
Timestamp: 12/10/2025, 22:10:04
Proof: 100
Previous Hash: 1
Transactions: 0


BlockFund
All Campaigns
Login
Register
All Active Campaigns
Discover and support amazing projects on our blockchain-powered crowdfunding platform
student Education
For the education of student
Target: 3000.0 coins
Raised: 100.0 coins
Progress: 3.0%
Deadline: 2025-10-31
View Details
student election
gunjan is leader
Target: 2000.0 coins
Raised: 2000.0 coins
Progress: 100.0%
Deadline: 2025-10-31
View Details


BlockFund
All Campaigns
Login
Register
student Education
For the education of student
Target: 3000.0 coins
Raised: 100.0 coins
Deadline: 2025-10-31
Support This Campaign
To make a donation, please login or create an account.
Register to Donate
Recent Donations
Advait_220 donated 100.0 coins
TX: b9665fb4776e80d6...



Technology Stack

- **Programming Languages**: Python (Flask backend), HTML/CSS/JavaScript (Frontend)
- **Frameworks/Libraries**: Flask, Jinja2, Werkzeug, SQLite3 (stdlib), Fetch API (frontend), Optional: web3.py for future chain integration
- **Blockchain/Consensus**: SHA-256 hashing, Proof-of-Work (configurable difficulty), Chain validation endpoints

GitHub Link: https://github.com/shantanudongare/blockfund/tree/main

YouTube Link: https://youtu.be/BfHEu93ZUno