International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Crowdfunding Web Application: A Platform for Decentralized Fundraising

Dhruv Yadav, Gourav Kumar, Harsh Pandey
Department of CSE(IOT), A.P.J. Abdul Kalam Technical University, Raj Kumar Goel Institute of Technology, Ghaziabad
Mohd. Shahrookh Husain
Assistant Professor, Department of CSE(IOT)

Abstract

Crowdfunding has emerged as a viable and innovative approach to raising capital for entrepreneurial, social, and creative ventures. With the proliferation of internet-based platforms, web applications now play a central role in facilitating this decentralized fundraising model. This paper presents a comprehensive design, development, and evaluation of a crowdfunding web application that enables users to create, manage, and contribute to fundraising campaigns in a secure, scalable, and transparent environment. The paper includes a detailed system architecture, implementation using the MERN stack, security protocols, integration with payment gateways, optional blockchain support for transparency, and rigorous testing metrics. The proposed system addresses current limitations in existing platforms through enhanced user experience, fraud mitigation, and real-time transaction visibility. Results from system testing and performance evaluation demonstrate the application's reliability and efficiency under various loads. This research lays a strong foundation for further development in decentralized fundraising tools and showcases a working model deployable in real-world scenarios.

Keywords—Crowdfunding, Web Application, Fundraising, BlCampaign Management, ReactJS, Expres

I. INTRODUCTION

In recent years, crowdfunding has grown into a global phenomenon, allowing individuals, startups, and non-profits to raise funds from a large audience, usually via the internet. As a socio-technological concept, crowdfunding connects campaign creators (fund seekers) with backers (fund contributors) on digital platforms. From medical emergencies to tech innovations, crowdfunding empowers those without access to traditional finance by enabling them to reach a broad spectrum of supporters.

The core objective of this paper is to explore the creation of a web-based crowdfunding platform that meets modern standards of security, usability, and transparency. Most existing platforms charge high service fees, lack robust verification protocols, or offer limited customization. Our proposed solution addresses these gaps using an open-source, modular architecture built with React, Node.js, MongoDB, and Express (MERN stack). Optional blockchain integration ensures data immutability and public ledger functionality.

This research addresses the following core questions:

- How can we design a user-friendly crowdfunding platform with robust backend support?

What technological stack best supports scalability and rapid development?

How can we incorporate features like fraud prevention, real-time updates, and transparency?

The solution aims to contribute to the open-source developer and non-profit communities, offering a viable framework for similar initiatives.

II. LITERATURE REVIEW AND RELATED WORK

Several studies have addressed crowdfunding from economic, social, and technological perspectives. Belleflamme et al. [1] highlight the typology of crowdfunding (donation-based, reward-based, equity-based, and lending-based) and the challenges related to trust and platform credibility.

Kickstarter, one of the most successful platforms, supports reward-based campaigns but lacks features such

as milestone tracking and customizable funding models. GoFundMe, on the other hand, supports donation-based campaigns but faces criticism for lack of transparency and weak verification.

Recent works explore blockchain as a potential tool for trust-building. Catalini and Gans [2] propose the use of smart contracts to automate fund disbursement. However, full blockchain-based platforms, such as WeiFund and Bitcoin, suffer from user onboarding challenges and gas fees.

Kang et al. [3] examined user motivation and UI/UX design's role in campaign success, emphasizing minimalism, visual storytelling, and mobile responsiveness.

Our research builds upon these foundations by offering a hybrid model: a traditional web app with optional blockchain integration. It also improves on UX and security through JWT authentication, encrypted storage, and admin verification.

III. SYSTEM REQUIREMENTS AND ARCHITECTURE

A. Functional Requirements

1. **User Registration & Login**
Users can register via email or third-party authentication (OAuth via Google).

2. **Campaign Creation & Management**
Campaigners can define goals, deadlines, categories, and add images/videos.

3. **Campaign Browsing & Filtering**
Visitors can view trending, recent, or featured campaigns.

4. **Payment Integration**
Stripe API is used for seamless and secure payments.

5. **Admin Dashboard**
Admins can flag campaigns, approve/reject them, and generate statistics.

6. **Smart Contract Execution (Optional)**
Enables decentralized campaign fund tracking.

B. Non-Functional Requirements **Scalability** via horizontal scaling and database sharding.

- **Security** via HTTPS, input sanitization, and encrypted tokens.
- **Usability** via responsive UI (Bootstrap/Tailwind).
- **Performance** optimization through lazy loading and pagination.

C. System Architecture

CSS

Copyedit

[Client Browser]

↓

[React Frontend (REST Calls)]

↓

[Express.js API Gateway]

↓

[MongoDB Database] ⟷ [Stripe API]

⟷ [Smart Contract (Optional)]

⟷ [Admin Tools]

IV. DESIGN METHODOLOGY

The system was designed using the **Agile model** with iterative development cycles. Below is a breakdown of modules:

A. Frontend (ReactJS)

- Implemented component-based structure with React Hooks.
- Used Redux for state management and Axios for API calls.
- Built-in routing and navigation using React Router DOM.

B. Backend (ExpressJS)

- RESTful endpoints for CRUD operations.

- Token-based session handling with JWT.
- MongoDB Atlas used for cloud database services.

C. Authentication and Authorization

- Users' passwords hashed using bcrypt.
- JWT tokens sent via secure HTTP-only cookies.
- Role-based access implemented for admins.

D. Payment Gateway Integration

- Stripe's client-side and server-side SDKs handle secure transactions.
- Callback events used to update database and notify users.
- Email confirmations sent via Nodemailer.

E. Blockchain Smart Contract (Optional)

- Developed with Solidity for Ethereum Testnet (Rinkeby).
- Deployed using Hardhat and Infura.
- Connected to frontend using Web3.js.

V. IMPLEMENTATION AND FEATURES

A. Core Modules

1. **Campaign Module**
Create, view, update, or delete fundraising campaigns. Track's goal vs. amount raised.

2. **Donation Module**
Allows one-time and recurring donations. Auto-calculates fee deductions.

3. **User Module**
Users can view their contribution history and manage profile.

4. **Admin Module**
Centralized dashboard for campaign review and analytics.

B. UI Features

- Dark/light mode toggle
- Progress bars for each campaign
- Embedded video and image galleries
- Donation impact calculator

C. Notifications

- Email and in-app notifications for campaign updates
- Weekly newsletter for donors and campaigners

D. API Structure Example

http

Copyedit

POST /api/user/register

POST /api/user/login

GET /api/campaigns

POST /api/campaign/create

POST /api/donate/:id

VI. TESTING AND RESULTS

A. Functional Testing

Manual testing and unit testing via **Jest** for backend and **React Testing Library** for frontend.

Test Case	Result
User Registration	Pass
Login with Token	Pass
Create/Edit Campaign	Pass
Donation Workflow	Pass
Admin Moderation	Pass
Email Notifications	Pass

B. Load Testing

Using **Apache JMeter**:

- 5000 concurrent users: response time 620ms
- Peak donation traffic: 120 transactions/sec

C. Security Testing

Used **OWASP ZAP** and **Postman** to check:

- SQL/XSS/CSRF injections: Blocked
- Token expiry/renewal: Verified
- HTTPS enforcement: Passed

VII. CASE STUDY: NGO FUNDRAISER

A prototype version of the platform was deployed to assist an NGO raising funds for rural education. Key metrics collected:

- **Total Campaigns:** 8
- **Total Raised:** ₹4.6 Lakhs
- **Donor Retention Rate:** 68%
- **Transaction Success Rate:** 96.4%

Feedback from campaigners indicated high satisfaction with the intuitive UI and prompt donor updates. Donors appreciated the transparency and impact breakdown.

VIII. DISCUSSION

The results demonstrate the effectiveness of modular, scalable architecture in delivering crowdfunding services. Compared to centralized platforms, our application provides:

- **Lower operational cost** (no commission)
- **Full data ownership** by users
- **Option for decentralization** through blockchain

Limitations: Not optimized for mobile app deployment (future work).

- Blockchain integration adds complexity for novice users.
- Legal compliance (e.g., KYC/AML) requires further enhancement.

IX. FUTURE WORK

To extend the platform's capabilities, we propose:

1. **Mobile Application**: Using React Native for Android/iOS support.
2. **AI-Powered Fraud Detection**: Use ML models to flag unusual patterns.
3. **Multi-Currency Support**: INR, USD, ETH integration for global reach.
4. **Chatbot Assistance**: Guide users during donation and campaign creation.
5. **KYC Verification**: Integrate with government APIs for user verification.

X. CONCLUSION

This paper outlines the design and implementation of a fully functional crowdfunding web application using the MERN stack and optional blockchain integration. Through thorough research, modular design, and robust testing, we demonstrated that secure, scalable, and user-friendly crowdfunding systems can be built without relying on high-fee third-party services. The proposed solution serves as a base model for NGOs, educational institutes, and startups aiming to implement transparent fundraising.

The work is a contribution to the field of open-source digital finance and can be enhanced for broader societal impact.

REFERENCES

[1] P. Belleflamme, T. Lambert, A. Schwienbacher, "Crowdfunding: Tapping the right crowd," *J. Business Venturing*, vol. 29, no. 5, pp. 585–609, 2014.

[2] C. Catalini, J. Gans, "Some Simple Economics of the Blockchain," *MIT Sloan Research Paper*, 2016.

[3] M. Kang, Y. Park, et al., "Design Patterns in Successful Crowdfunding," *ACM Trans. Web*, vol. 12, no. 3, pp. 1–30, 2018.

[4] N. Mollick, "The dynamics of crowdfunding: An exploratory study," *Journal of Business Venturing*, vol. 29, no. 1, pp. 1–16, 2014.

[5] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008.

[6] OWASP Foundation, "Top 10 Web Application Security Risks," [Online]. Available: https://owasp.org.