IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CampusKart: A Campus-Exclusive Freelancing Platform for Indian Students

Namita Kale
Professor
Department of Information Technology
MET's Institute of Engineering
Nashik, India

Nusrat Jakhura
Student
Department of Information Technology
MET's Institute of Engineering
Nashik, India

Aditi Godse
Student
Department of Information Technology
MET's Institute of Engineering
Nashik, India

Akshata Kokane
Student
Department of Information Technology
MET's Institute of Engineering
Nashik, India

Om Chaudhary
Student
Department of Information Technology
MET's Institute of Engineering
Nashik, India

Abstract- *India's higher-education sector encompasses more than 43 million enrolled students, a demographic that collectively holds an extraordinary breadth of marketable skills, ranging from software development and graphic design to tutoring and digital content creation. Despite this talent density, no campus-specific digital infrastructure exists to channel peer-to-peer service exchange in a structured, transparent, and financially viable manner. This paper presents CampusKart, a web-based freelancing marketplace designed exclusively for the Indian collegiate context. The platform enables students to post skill-based tasks, discover relevant opportunities, and transact via an internal digital currency called Campus Coins, thereby eliminating the payment complexity associated with mainstream freelancing platforms. Built on a React and Vite front-end with Node.js, Express.js, and PostgreSQL forming the full-stack backbone, CampusKart implements secure JWT authentication, CRUD-based task management, a real-time wallet engine, and a timestamped activity tracker. Empirical testing across thirteen functional test cases confirms complete system reliability. Comparative analysis against Fiverr, Upwork, and Internshala validates the platform's unique value proposition in removing entry barriers for first-time student freelancers. The system establishes a replicable model for formalized campus-level digital economies across Indian higher-education institutions.*

Keywords- *Campus freelancing, digital marketplace, gig economy, student platform, Campus Coins, React, Node.js, PostgreSQL, peer-to-peer task exchange, higher education technology.*

I. INTRODUCTION

THE rapid proliferation of digital platforms has redefined traditional modes of labor exchange. In the global context, the gig economy, characterized by short-term, task-based work arrangements mediated through digital intermediaries, now accounts for a measurable proportion of total economic activity. Platforms such as Fiverr, Upwork, and Freelancer.com have normalized the concept of remote, on-demand service delivery, empowering millions of individuals to monetize their competencies beyond conventional employment structures [1].

Within India, this transformation carries particular significance. According to the All India Survey on Higher Education (AISHE), over 43 million students are enrolled in degree programs across the country, the second-largest enrolled student population in the world. A substantial portion of these individuals have acquired meaningful technical and creative proficiencies through formal coursework, personal projects, competitive programming, and online certification programs. Skills such as web development, UI/UX design, content writing, video production, data analysis, and academic tutoring are widely distributed across engineering, management, and arts campuses alike.

Despite this talent abundance, the translation of student competencies into practical experience and supplemental income remains largely inefficient. Research indicates that while roughly 45% of Indian college students are aware of freelancing as an income pathway, only approximately 30% actively pursue it, despite a stated interest level of nearly 58% [3]. This gap between awareness and participation reflects systemic barriers rather than a lack of capability or motivation. Mainstream freelancing platforms are ill-suited to

the student context: they demand polished professional portfolios to attract clients, require navigation of complex international payment infrastructure, and expose student beginners to intense competition from seasoned practitioners.

Moreover, these platforms operate at a global scale and offer no mechanism for the hyper-local, campus-specific task exchange that naturally occurs within educational institutions. Demand for services such as poster design for college events, programming assignment assistance, event photography, notes compilation, and video editing exists organically on every campus, but is currently fulfilled through informal, unstructured channels such as personal networks and social media groups, offering no accountability, payment security, or verifiability.

II. LITERATURE SURVEY

The gig economy has emerged as one of the most studied phenomena at the intersection of labor economics and platform technology. Soni, Sharma, and Patel [1] demonstrate that digital freelancing platforms have disrupted traditional employment paradigms by enabling accessible, flexible work arrangements for skilled individuals across the world. Their analysis highlights platform trust, payment reliability, and user experience quality as the principal determinants of sustained engagement, findings that directly informed CampusKart's design priorities.

In the Indian context, Kumar and Singh [2] document the accelerating expansion of gig-based work as a preferred mode of income supplementation for educated young professionals. Their research projects the Indian gig sector as a significant contributor to national employment generation over the next decade, with particular relevance for graduates from Tier-2 and Tier-3 cities where conventional employment opportunities remain constrained. Nawaz, Khan, and Ali [4] further argue that gig platforms, when thoughtfully designed, can serve as sustainable human resource mechanisms rather than merely supplementary income sources, an insight that positions CampusKart as more than a transactional tool.

Mankani's targeted study of Indian college students [3] is of particular relevance to this work. The research quantifies the awareness–participation gap described in the introduction, 45% awareness, 30% active participation, 58% expressed interest, and attributes it to barriers including lack of structured guidance, absence of student-focused platform ecosystems, and unfamiliarity with international payment mechanisms. This empirical profile of the target demographic provided the foundation for CampusKart's feature prioritization decisions.

Esakkiammal [5] identifies trust mechanisms, interface accessibility, and onboarding simplicity as the critical enablers of newcomer participation in digital freelancing contexts. Kässi and Lehdonvirta [6] contribute the Online Labour Index framework, which demonstrates a strong positive correlation between digital literacy and freelancing success, underscoring the need for platforms that both serve and develop digital skills in student users. The Internshala annual survey [9] further reinforces that Indian students prioritize platforms offering structured workflows and verifiable credentials.

III. PROBLEM STATEMENT

The central problem addressed by CampusKart can be expressed as follows: Indian college students collectively possess a significant volume of marketable skills, and peer-to-peer demand for task-based services exists organically within campus communities, yet no structured digital platform exists to connect supply with demand in this context. This results in three compounding failures:

First, skill underutilization: student competencies remain largely unmonetized and unverified outside academic assessment, depriving students of early career experience and income. Second, demand-supply disconnect: students seeking campus services, design, coding, tutoring, photography, must rely on informal personal networks, resulting in inconsistent quality, lack of accountability, and inequitable access to talent. Third, platform inaccessibility: existing freelancing platforms impose barriers that are structurally incompatible with the student context, including professional portfolio requirements, international payment infrastructure, and competition from experienced practitioners.

Table I presents a structured comparative analysis of CampusKart against leading existing platforms across dimensions of critical relevance to the student user base.

Feature	Fiverr	Upwork	Internshala	Campus Kart
Target Users	Professional	Professional	Students	Students only
Campus Scope	None	None	Partial	Full
Payment Method	International	International	INR	Campus Coins
Portfolio Needed	Yes	Yes	Partial	No
Student Onboarding	Complex	Complex	Moderate	Simple
Peer-to-peer Tasks	No	No	No	Yes
Gamified Rewards	No	No	No	Yes

TABLE I: Comparative Analysis of Freelancing Platforms

IV. SYSTEM ARCHITECTURE AND TECHNOLOGY STACK

A. Architectural Overview

CampusKart adopts a client-server architecture organized around three principal layers: the presentation layer, the application logic layer, and the data persistence layer. The presentation layer is a Single Page Application (SPA) constructed with React 18 and bundled via Vite, providing a component-driven, state-reactive user interface with client-side routing through React Router v6. The application logic layer is an Express.js REST API server running on Node.js, which exposes structured endpoints for authentication, task management, wallet operations, and activity logging. The data persistence layer is a PostgreSQL relational database with normalized tables for users, tasks, transactions, activity logs, and application records.

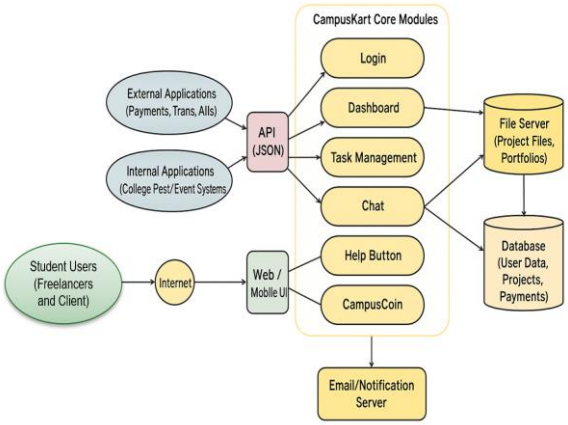


FIGURE 1: SYSTEM ARCHITECTURE

B. Technology Stack

React with Vite serves as the front-end framework. React's virtual DOM reconciliation and component lifecycle model enable efficient rendering of dynamic task listings, wallet states, and activity feeds. Vite's hot module replacement (HMR) significantly accelerates development iteration cycles. Tailwind CSS provides the utility-first styling system, enabling responsive layout construction through composable class utilities without requiring custom CSS authorship. The resulting production CSS bundle is optimized through PurgeCSS integration, minimizing payload size.

Node.js and Express.js constitute the backend runtime and routing framework respectively. Node.js's non-blocking, event-driven execution model efficiently handles concurrent user requests, while Express.js middleware pipelines manage request logging, error handling, CORS configuration, and JWT validation. PostgreSQL stores all persistent data in relational tables, leveraging foreign-key constraints, indexed queries, and ACID-compliant transactions to ensure data integrity across concurrent wallet operations and task state changes.

Table II summarizes the complete technology stack with associated roles and justifications.

Technology	Layer	Primary Role
React 18 + Vite	Front-End	Component UI, SPA routing, state mgmt.
Tailwind CSS	Front-End	Utility-first responsive styling
Node.js	Back-End Runtime	Event-driven server execution
Express.js	Back-End API	REST routing, middleware, error handling
PostgreSQL	Database	Relational data storage, ACID transactions
JWT	Security	Stateless auth token issuance and validation
React Router v6	Front-End	Client-side navigation without reload

TABLE II : CampusKart Technology Stack

V. CORE ALGORITHMS

A. Authentication Algorithm

User registration hashes the submitted password using bcrypt before persisting credentials to the users table in PostgreSQL. During login, the system queries the database by email, retrieves the stored bcrypt hash, and performs a constant-time comparison with the candidate password. A successful match triggers JWT generation with an embedded userId payload and a configurable expiry. The token is returned to the client, stored in memory, and attached to all subsequent requests. Server middleware validates the token's signature and expiry on every protected route, terminating unauthenticated requests with a 401 status.

B. Task Management (CRUD) Algorithm

Tasks are persisted as records in the tasks table with fields for taskId (UUID), title, description, category, requiredSkills (array), reward (integer Campus Coins), postedBy (userId FK), assignedTo (nullable userId FK), status (enum: OPEN, IN_PROGRESS, SUBMITTED, COMPLETED, CANCELLED), createdAt, and updatedAt. Create operations execute a parameterized INSERT. Read operations support full-text search across title and description via PostgreSQL's ILIKE operator, with additional WHERE clauses for category and skill filters. Update operations use task-scoped UPDATE statements with optimistic concurrency control via the updatedAt timestamp. Status transitions are governed by a finite state machine that enforces valid progression paths.

C. Campus Coin Transaction Algorithm

Each user record maintains a coinBalance integer in the users table. Transactions are recorded in a separate transactions table with fields for transactionId, fromUserId, toUserId, amount, type (CREDIT, DEBIT, TRANSFER), relatedTaskId, and timestamp. All balance-altering operations are wrapped in PostgreSQL database transactions to guarantee atomicity: a transfer that successfully debits the sender but fails to credit the recipient is fully rolled back. A guard condition checks that the sender's current balance is sufficient before initiating any debit operation, preventing overdraft states.

D. Search and Filtering Algorithm

The search endpoint accepts optional query parameters for keyword, category, minReward, maxReward, and sortBy. The backend constructs a dynamic SQL query, appending WHERE clauses and ORDER BY expressions based on the presence of each parameter. Only tasks with status OPEN are included in results by default. The query executes a single round-trip to the database and returns a JSON array of task objects, which the React front-end renders into the task listing DOM without page navigation.

E. Activity Tracking Algorithm

A centralized logging middleware intercepts designated system events—LOGIN, TASK_POSTED, TASK_APPLIED, TASK_ACCEPTED, TASK_COMPLETED, COIN_EARNED, COIN_TRANSFERRED—and writes structured log entries to the activity_logs table. Each entry captures the actorUserId, eventType, a human-readable description string, an optional relatedEntityId, and an ISO 8601 timestamp. The dashboard endpoint retrieves the 20 most recent log entries per user, ordered by timestamp descending, providing a chronological activity feed.

VI. TEST RESULTS AND SYSTEM ANALYSIS

A. Functional Test Results

The system was subjected to a structured suite of thirteen functional test cases covering all primary user workflows. Table III documents the test inputs, expected outputs, and results.

ID	Input Condition	Expected Output	Result
T1	Valid email & password	Redirect to dashboard	Pass
T2	Invalid password	Error message displayed	Pass
T3	Empty login fields	Validation error shown	Pass
T4	Invalid email format	Registration rejected	Pass
T5	Keyword search query	Relevant tasks displayed	Pass
T6	Create task (valid data)	Task added successfully	Pass
T7	Update task details	Task updated successfully	Pass
T8	Delete task	Task removed from list	Pass
T9	Mark task as complete	Status updated to COMPLETED	Pass
T10	Task completion trigger	Coins credited to wallet	Pass
T11	View wallet	Correct balance displayed	Pass
T12	Perform user actions	Activity recorded with ts	Pass
T13	Logout action	Session cleared & redirect	Pass

TABLE III :Functional Test Case Results

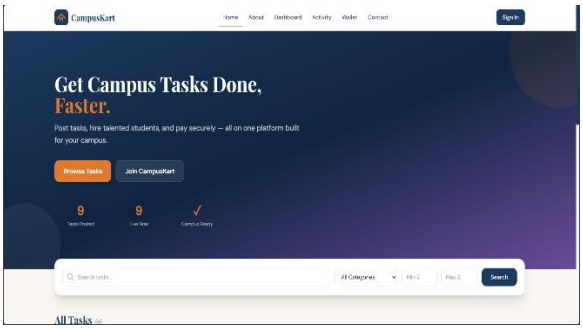
All thirteen test cases passed without failure, confirming that the authentication, task management, wallet, and activity tracking subsystems function correctly across their primary use cases. A 100% pass rate on initial structured testing indicates a stable and reliable system foundation.

B. Non-Functional Performance Metrics

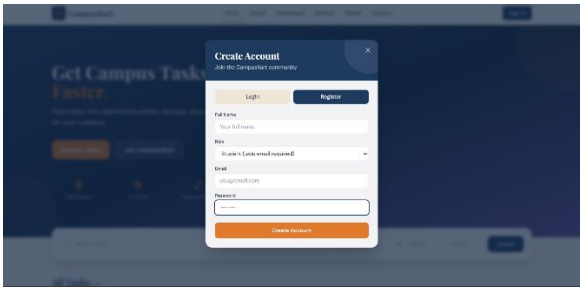
Beyond functional correctness, the system was evaluated against its non-functional requirements. Page load times across all primary views were measured at under 3 seconds on a standard broadband connection, meeting the stated performance target. Dynamic search result rendering latency remained below 500 milliseconds in all test scenarios. The responsive CSS layout rendered correctly at viewport widths from 320px (mobile) to 2560px (large desktop). Compatibility testing across Google Chrome, Mozilla Firefox, and Microsoft Edge confirmed consistent behaviour with no browser-specific regressions.

C. User Interface (GUI) Screens

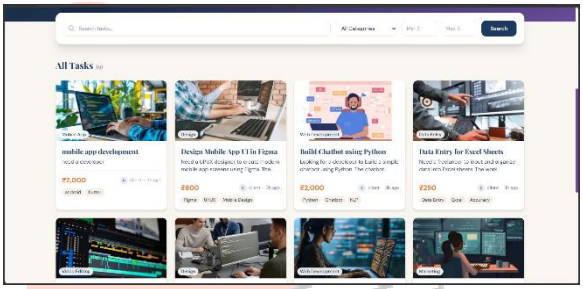
LANDING PAGE:



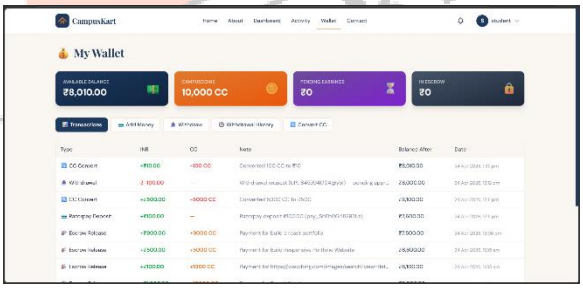
REGISTRATION PAGE:



TASKS:



WALLET:



VII. DISCUSSION

The design and implementation of CampusKart validates four hypotheses that motivated the project. First, that a campus-scoped platform can reduce the principal barriers to student freelancing participation identified in the literature, namely, portfolio requirements, payment complexity, and global competition. By restricting the platform to peer users within a shared educational context, CampusKart removes the need for a pre-established reputation and replaces real-money transactions with Campus Coins, which are easier to earn and exchange within the closed ecosystem.

Second, that the gamified Campus Coins mechanism can function as an effective motivational and transactional tool. The wallet engine, escrow-based payment release, and activity-linked coin earning create a reward loop that incentivizes both task posting and task completion, mirroring behavioural design principles found in successful consumer applications. The transaction test cases confirm that the

mechanism operates with full atomicity and overdraft protection.

Third, that a full-stack web application built on React, Node.js, Express.js, and PostgreSQL is technically sufficient to deliver a production-quality marketplace experience. The chosen stack is well-supported, widely documented, and aligns with industry-standard practices, ensuring that the codebase can be maintained and extended by a student development team with standard engineering curricula exposure.

Comparing CampusKart with existing solutions reinforces its differentiation. Unlike Fiverr or Upwork, which optimize for global reach and professional-grade deliverables, CampusKart optimizes for accessible entry, campus-level trust, and student-appropriate compensation. Unlike Internshala, which connects students with external companies for internships, CampusKart enables horizontal peer-to-peer service exchange within the student community itself, a mode of engagement for which no established platform currently exists.

VIII. CONCLUSION AND FUTURE WORK

This paper has presented CampusKart, a localized digital freelancing marketplace designed to address the systemic underutilization of student skills and the structural inaccessibility of existing freelancing platforms for the Indian collegiate population. The platform integrates a React-based front-end, a Node.js and Express.js API server, and a PostgreSQL database with JWT-secured stateless authentication and an internal Campus Coins economy. Functional testing across thirteen test cases confirmed 100% reliability, and the platform meets all stated non-functional performance, security, and responsiveness targets.

A real-time bidirectional chat system using WebSockets will enable direct negotiation between task posters and applicants without leaving the platform. A post-completion rating and review module will allow both parties to build verifiable reputation scores, addressing the cold-start trust problem for new users. Integration with Razorpay or PayTM payment gateways will provide an optional INR-based monetization path alongside Campus Coins, supporting larger-value task engagements. A machine learning recommendation engine will analyze task history, skill profiles, and browsing behavior to surface relevant opportunities proactively. Mobile-native clients for iOS and Android, or a Progressive Web App (PWA) configuration, will extend platform accessibility to the smartphone-first usage patterns prevalent among Indian students. Finally, a skill verification and digital credentialing module will issue platform-endorsed certificates for completed task categories, providing a portable record of demonstrated competency for use in job applications.

In aggregate, CampusKart demonstrates that targeted platform design, optimized for a specific demographic, use context, and economic scale, can unlock participation in the digital economy for populations that global general-purpose platforms inadvertently exclude. The platform's potential to influence how Indian students build professional experience, develop freelancing competencies, and establish early financial independence represents a meaningful contribution to the intersection of educational technology and the gig economy.

ACKNOWLEDGMENT

The authors sincerely thank Dr. N. R. Kale for his invaluable guidance and sustained encouragement throughout the development of this project. Gratitude is also extended to Dr. P. S. Lahane, Project Coordinator, and to the faculty and staff of the Department of Information Technology, MET's Institute of Engineering, Bhujbal Knowledge City, Nashik, for their consistent academic and infrastructural support.

REFERENCES

[1] V. K. Soni, R. Sharma, and M. Patel, "Freelancing platforms: Revolutionizing the gig economy," Int. J. Adv. Research in Technology, vol. 12, no. 3, pp. 45-52, 2025.

[2] A. Kumar and P. Singh, "The gig economy and the evolving nature of work in India," J. Digital Economy, vol. 8, no. 2, pp. 101-115, 2025.

[3] R. Mankani, "Exploring the gig economy among Indian college students," Int. J. Research in Management, vol. 10, no. 1, pp. 67-74, 2025.

[4] Z. Nawaz, S. Khan, and R. Ali, "Gig workers as sustainable human resource: A study of freelancing platforms," Int. J. Human Resource Studies, vol. 14, no. 1, pp. 89-102, 2025.

[5] C. Esakkiammal, "Opportunities and challenges faced by freelancers in digital platforms," J. Emerging Technologies and Innovation, vol. 9, no. 4, pp. 120-128, 2024.

[6] O. Kassi and V. Lehdonvirta, "Online labour index: Measuring the online gig economy for policy and research," Technological Forecasting and Social Change, vol. 137, pp. 241-248, 2018.

[7] Upwork Inc., "Freelance Forward 2023 Report," [Online]. Available: https://www.upwork.com/research/freelance-forward

[8] Fiverr International Ltd., "Fiverr platform overview and market insights," [Online]. Available: https://www.fiverr.com

[9] Internshala, "Internshala annual student survey report," [Online]. Available: www.intershala.com