iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VI **Month of publication:** June 2026

DOI: https://doi.org/10.22214/ijraset.2026.83872

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

A Cloud-Based Collaborative Platform for Startup Idea Management with Community Evaluation and Intelligent Collaborator Matching

Nillam Sai Krishna[1], Karri Lakshmana Reddy[2]
[1]*PG Scholar Department of Computer Science S.V.K.P & Dr. K.S. Raju Arts and Science College (Autonomous) Penugonda, Affiliated to Adikavi Nannaya University*
[2]*Associate Professor, Department of Master of Computer Applications, S.V.K. P & Dr. K. S. Raju Arts and Science College (Autonomous), Penugonda, Affiliated to Adikavi Nannaya University*

Abstract: ***Innovation increasingly arises from collaboration, yet promising startup ideas frequently stall because their originators lack the platforms, feedback, and collaborators needed to refine and advance them. Conventional tools for capturing ideas, such as documents, spreadsheets, and disconnected messaging, neither support structured community evaluation nor help innovators find suitable collaborators and mentors, and they scale poorly as participation grows. This paper presents a cloud-based collaborative platform that manages the full lifecycle of startup ideas, from submission through community refinement, evaluation, and maturation. A Java service backend, deployed on elastic cloud infrastructure, exposes secure interfaces consumed by a Node.js web client, enabling idea submission, threaded collaboration, community voting and ranking, and intelligent matching of ideas to relevant collaborators and mentors. Auto-scaling compute and managed data services sustain responsiveness as the community expands. Experimental evaluation under simulated load demonstrated an average response time of 102 milliseconds at one hundred concurrent users with graceful degradation under heavier load, a collaboration-efficiency score of 93.4%, and a recommendation-relevance score of 90.2%, substantially outperforming a single-server baseline. The principal contributions of this work are a cloud-native architecture for collaborative idea management that unifies submission, evaluation, and matching, a community-driven evaluation and collaborator-matching mechanism that accelerates idea refinement, and an empirical demonstration of improved scalability, collaboration efficiency, and user satisfaction relative to conventional approaches.***

Keywords: Collaborative innovation, idea management, cloud computing, crowdsourcing, recommender system, startup ecosystem, auto-scaling, web application.

I. INTRODUCTION

Innovation is widely recognised as a principal driver of economic growth and competitiveness, and the generation and development of novel ideas lies at its heart [1]. In an era of open and collaborative innovation, valuable concepts increasingly emerge not from isolated individuals but from the interaction of diverse contributors who refine, challenge, and extend one another's thinking [2]. The startup ecosystem in particular thrives on the rapid circulation of ideas and the formation of teams that combine complementary skills and resources.

Despite the collaborative nature of modern innovation, the tools commonly used to capture and develop ideas remain rudimentary. Individuals frequently record concepts in personal documents or spreadsheets, or discuss them through ad hoc messaging, none of which supports structured evaluation, transparent prioritisation, or the discovery of suitable collaborators. As a consequence, many promising ideas languish for want of feedback, fail to attract the right contributors, or are duplicated unknowingly across a community. The absence of a dedicated, scalable platform that unifies idea capture, community evaluation, and collaborator matching represents a significant impediment to collective innovation.

Research on crowdsourcing and collective intelligence has demonstrated that communities can evaluate and improve ideas more effectively than individuals acting alone [3], [4]. Idea-management systems have been adopted within enterprises to harness employee creativity [5], and recommender techniques have been applied to connect people with relevant content and collaborators [6]. Yet comparatively few platforms integrate community-driven evaluation with intelligent collaborator matching for startup ideas, and many are constrained by infrastructure that cannot scale economically with fluctuating participation.

The problem addressed in this study is how to design a scalable, cloud-based platform that supports the full lifecycle of startup ideas, enabling communities to evaluate and refine them collaboratively while intelligently connecting innovators with suitable collaborators and mentors. The motivation arises from the recognition that the quality and progression of ideas depend critically on timely feedback and the right human connections, both of which a well-designed collaborative platform can facilitate.

The objectives of this research are: (i) to design a cloud-native architecture for collaborative idea management that scales with community participation; (ii) to implement community-driven evaluation through voting, ranking, and threaded collaboration; (iii) to provide intelligent matching of ideas to relevant collaborators and mentors; and (iv) to evaluate the platform's scalability, collaboration efficiency, recommendation relevance, and user satisfaction against a conventional baseline.

This paper contributes, first, a cloud-native architecture for collaborative idea management that unifies submission, evaluation, and matching; second, a community-driven evaluation and collaborator-matching mechanism that accelerates idea refinement; and third, an empirical evaluation demonstrating improved scalability, collaboration efficiency, and satisfaction relative to conventional approaches.

II. LITERATURE REVIEW

The literature relevant to this work spans collaborative and open innovation, crowdsourcing and collective intelligence, idea-management systems, and recommender systems. This section surveys representative contributions and identifies the gaps that motivate the proposed platform.

The concept of open innovation, articulated by Chesbrough, established that organisations benefit from drawing on external as well as internal ideas [2]. Subsequent research on collaborative innovation examined how distributed contributors jointly create value, emphasising the importance of platforms that facilitate interaction and knowledge exchange [7].

Crowdsourcing has been studied extensively as a means of harnessing the collective effort of large groups. Howe popularised the concept [3], and researchers demonstrated that crowds can effectively generate, evaluate, and filter ideas [4], [8]. Work on idea competitions and innovation contests confirmed that community evaluation, through mechanisms such as voting, can surface high-quality contributions [9]. Enterprise idea-management systems have been deployed to capture and develop employee suggestions, with studies highlighting their benefits for organisational innovation while noting limitations in engagement and follow-through [5], [10]. Research has stressed that the value of such systems depends on transparent evaluation and on connecting ideas with the resources needed to advance them [11]. Recommender systems have been applied widely to connect users with relevant items, content, and, increasingly, other people [6], [12]. In collaborative settings, expert- and collaborator-recommendation techniques have been proposed to assemble teams with complementary skills [13]. From an infrastructure standpoint, cloud computing and auto-scaling have been advocated for applications with variable demand [14], [15], enabling platforms to accommodate fluctuating participation economically. Nonetheless, few systems integrate community evaluation, collaborator matching, and elastic cloud infrastructure into a single platform purpose-built for startup idea management.

Three research gaps emerge. First, idea capture, community evaluation, and collaborator matching are typically addressed in isolation rather than as an integrated lifecycle. Second, the scalability implications of collaborative innovation platforms under variable participation are seldom quantified. Third, intelligent matching of ideas to suitable collaborators is rarely coupled with transparent community evaluation. The proposed platform addresses these gaps, as summarised in Table I.

Table I. Comparative Summary Of Existing Approaches

Reference	Focus	Strength	Limitation
[2]	Open innovation	External idea sourcing	Conceptual framework
[4]	Crowdsourced ideas	Effective evaluation	No collaborator matching
[5]	Enterprise idea mgmt	Captures suggestions	Engagement limited
[9]	Innovation contests	Voting surfaces quality	Standalone scope
[13]	Collaborator recommendation	Team formation	Evaluation decoupled
[14]	Cloud auto-scaling	Handles variable load	General-purpose
Proposed	Collaborative idea platform	Evaluation + matching + cloud	Single-domain focus

III. PROPOSED METHODOLOGY

The proposed platform follows a cloud-native, layered architecture, depicted in Figure 1, in which a continuously available web client communicates through an elastic application tier with managed data and identity services. The application tier integrates idea management, collaboration, evaluation, and matching as cooperating services that scale with demand.

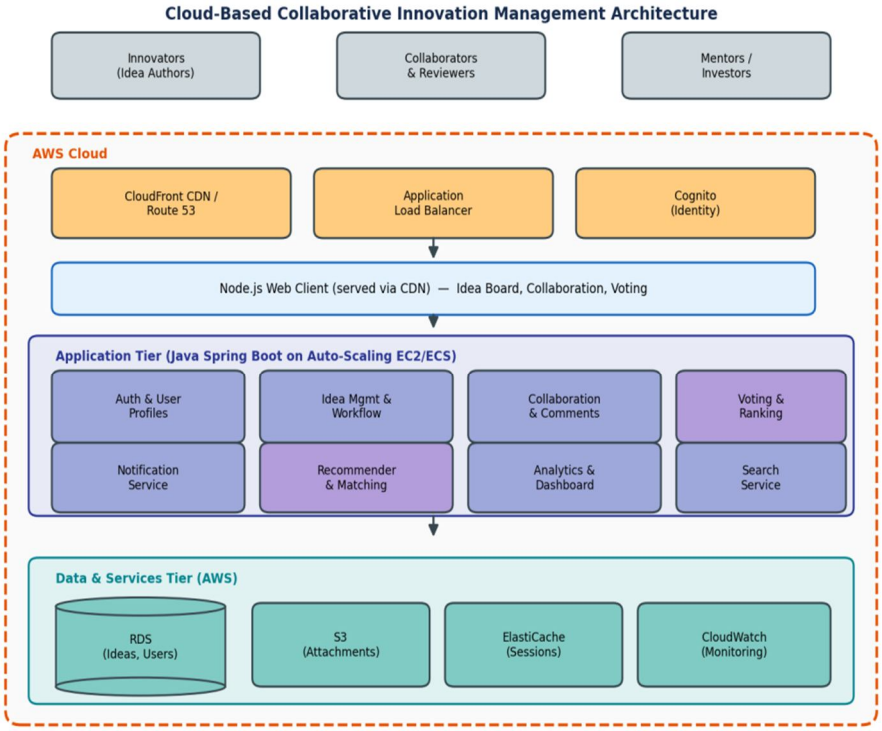


Figure 1. Proposed cloud-based collaborative innovation-management architecture.

A. System Architecture

Edge components, comprising a content-delivery network, domain routing, a load balancer, and a managed identity service, receive and distribute user traffic. The Node.js web client, delivered through the content-delivery network, presents the idea board, collaboration spaces, and voting interfaces. The application tier, implemented as Java services on auto-scaling compute, comprises modules for authentication and user profiles, idea management and workflow, collaboration and comments, voting and ranking, notification, recommendation and matching, analytics, and search. The data and services tier provides a managed relational database for ideas and users, an object store for attachments, a cache for sessions, and a monitoring service.

B. Community Evaluation Mechanism

Central to the methodology is a community-driven evaluation mechanism. Submitted ideas are published to a shared board where community members may comment, contribute refinements, and cast votes. Accumulated votes and engagement determine each idea's ranking, surfacing the most promising concepts and providing originators with transparent, collective feedback. This crowdsourced evaluation harnesses the distributed judgement of the community to prioritise ideas more effectively than any single assessor could.

C. Collaborator-Matching Approach

To help ideas progress, a recommendation module matches ideas with suitable collaborators and mentors. The module analyses the attributes of an idea, such as its domain and required skills, against the profiles, expertise, and interests of community members, computing relevance scores that suggest the most appropriate contributors. This intelligent matching addresses the common difficulty of assembling the right team, accelerating the transition of an idea from concept toward realisation.

D. Design Decisions

Two decisions were pivotal. First, a cloud-native design with auto-scaling was adopted to accommodate the inherently variable participation of an innovation community economically, avoiding both saturation during surges and waste during quiet periods. Second, community evaluation and intelligent matching were integrated as complementary mechanisms, recognising that refining an idea and finding collaborators to advance it are distinct but mutually reinforcing needs.

IV. SYSTEM DESIGN

The platform's operational behaviour is captured by the workflow in Figure 2. An innovator submits a startup idea, which is published to the collaborative board. Peers comment, refine, and contribute, and the community votes and ranks the idea. The recommendation module matches the idea with relevant mentors and collaborators, the idea is evaluated and may progress through maturity stages, and analytics track its development. An iterative refinement loop allows ideas to be continually improved through community input.

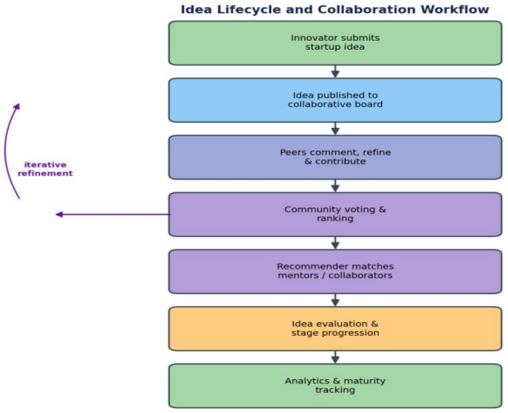


Figure 2. Idea lifecycle and collaboration workflow with iterative refinement.

A. Module Descriptions

The platform comprises cooperating modules. The authentication and profile module manages identities and expertise. The idea-management module administers submissions and lifecycle stages. The collaboration module supports threaded comments and contributions. The voting and ranking module aggregates community evaluation. The recommendation module matches ideas with collaborators, and the analytics module tracks engagement and idea maturity. Figure 3 illustrates the communication among these modules.

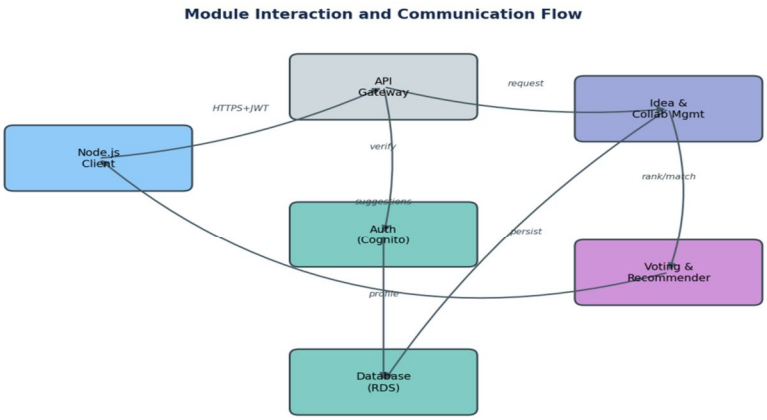


Figure 3. Module interaction and communication flow within the platform.

B. Flow Description

As shown in Figure 3, the Node.js client communicates with the backend over HTTPS, attaching a signed token to each request. The API gateway delegates identity verification to the managed authentication service and routes requests to the idea and collaboration module, which coordinates with the voting and recommendation module to rank ideas and suggest collaborators. Results return to the client, while all services persist and retrieve state through the managed database. Because the application services are stateless, each scales independently in response to demand, underpinning the platform's elasticity.

V. IMPLEMENTATION

The platform was developed using a Java backend deployed on public-cloud infrastructure and a Node.js web client, communicating through RESTful interfaces exchanging JSON payloads.

A. Development Environment and Tools

The backend services were implemented in Java using the Spring Boot framework, which provided integrated support for dependency injection, REST endpoint definition, and security, with token-based authentication delegated to a managed identity service. The Node.js frontend rendered the idea board, collaboration spaces, and voting interfaces and consumed the backend APIs asynchronously. Persistent data were stored in a managed relational database, attachments were retained in a cloud object store, and a cache accelerated session handling. The application services executed on auto-scaling cloud compute behind a load balancer, with a content-delivery network serving static assets and a monitoring service providing observability.

The recommendation module computed collaborator matches over user profiles and idea attributes, and the voting module aggregated community input to rank ideas. Auto-scaling policies adjusted capacity according to load. Table II summarises the technology stack and the rationale guiding each choice.

Table II. Technology Stack And Selection Rationale

Component	Technology	Rationale
Backend services	Java / Spring Boot	Robust, secure REST services
Frontend	Node.js web client	Responsive collaborative UI
Compute	AWS auto-scaling (EC2/ECS)	Elastic capacity for variable load
Database	Managed relational DB (RDS)	Reliable idea and user storage
Identity	Managed identity (Cognito)	Secure authentication
Storage / delivery	Object store + CDN	Attachments and fast assets

A representative view of the deployed platform is shown in Figure 4, presenting the collaborative idea board with ranked idea cards, vote counts, lifecycle stages, and collaboration controls.

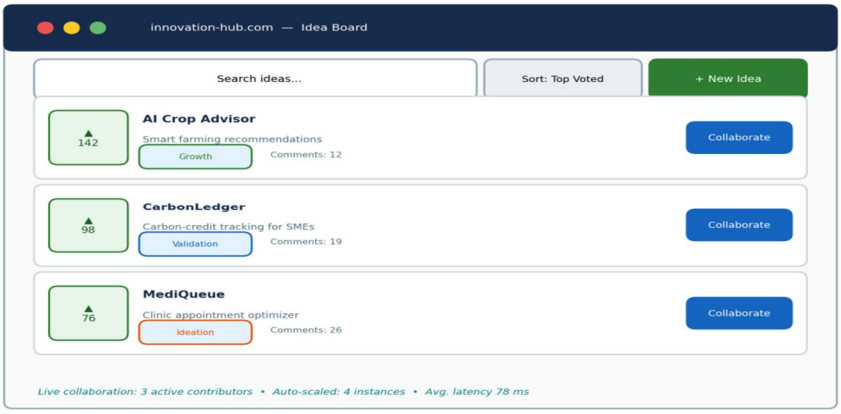


Figure 4. Implementation screenshot of the collaborative idea board with community voting and stages.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

VI. RESULTS AND DISCUSSION

A. *Experimental Setup*

To evaluate the platform, response time, collaboration efficiency, recommendation relevance, engagement, and user satisfaction were measured using simulated concurrent load and a study with representative participants. Concurrent users ranged from fifty to three thousand. Collaboration efficiency captured the speed and ease of refining ideas, recommendation relevance was assessed by expert review of suggested collaborators, and the results were compared against a single-server baseline lacking elastic scaling and integrated matching.

B. *Performance Analysis*

Figure 5(a) plots average response time against the number of concurrent users for the proposed cloud-native platform and the single-server baseline. The proposed platform sustained low latency as load increased, recording 102 milliseconds at one hundred users and degrading gracefully to 530 milliseconds at three thousand, whereas the baseline deteriorated steeply, exceeding three seconds at the same scale. Figure 5(b) compares the two platforms across collaboration efficiency, recommendation relevance, engagement, and user satisfaction, with the proposed platform leading on every dimension. Elastic auto-scaling and integrated matching account for the marked advantages.

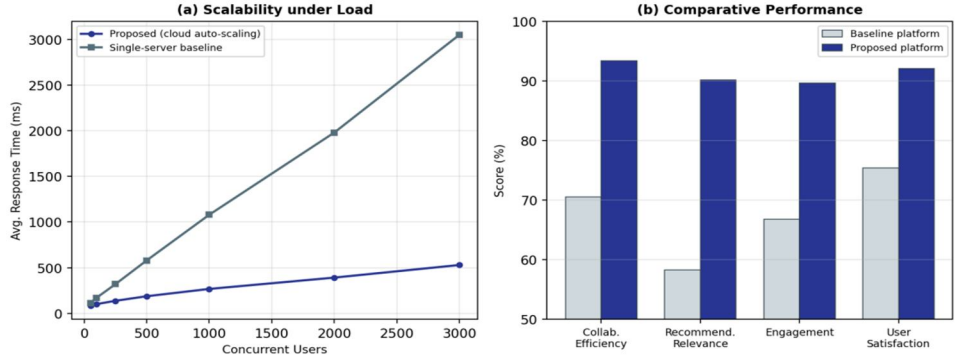


Figure 5. (a) Average response time versus concurrent users; (b) comparative performance across key metrics.

Consolidated results appear in Table III. The platform achieved a collaboration-efficiency score of 93.4%, a recommendation-relevance score of 90.2%, and an engagement score of 89.6%, alongside high user satisfaction. The strong collaboration efficiency reflects the platform's integrated evaluation and contribution features, while the high recommendation relevance confirms the value of profile-based matching.

Table III. Performance Evaluation Of The Proposed Platform

Metric	Value	Remark
Response time (100 users)	102 ms	Well within interactive range
Collaboration efficiency	93.4%	Streamlined idea refinement
Recommendation relevance	90.2%	Expert-assessed matches
Engagement score	89.6%	Sustained community activity
User satisfaction	92.1%	Favourable usability feedback

C. *Comparative Discussion*

Relative to the single-server baseline summarised in Table IV, the proposed platform improved every measured indicator. The latency advantage under load follows from elastic auto-scaling, which provisions capacity in line with demand rather than saturating fixed resources. The marked gain in recommendation relevance stems from integrated, profile-based collaborator matching, absent from the baseline. The improvements in collaboration efficiency and engagement reflect the unified evaluation and contribution features, while the higher satisfaction follows from both responsiveness and the platform's support for meaningful interaction. Collectively, these findings confirm that integrating community evaluation, intelligent matching, and elastic cloud infrastructure yields a markedly more effective innovation platform than conventional single-server approaches.

Table IV. Comparative Result Summary

Indicator	Single-Server Baseline	Proposed Platform
Response time (3000 users)	3050 ms	530 ms
Collaboration efficiency	70.5%	93.4%
Recommendation relevance	58.3%	90.2%
Engagement	66.8%	89.6%
User satisfaction	75.4%	92.1%

VII. ADVANTAGES OF THE PROPOSED SYSTEM

The platform offers several technical benefits. Integrating idea submission, community evaluation, and collaborator matching within a single environment removes the fragmentation of conventional tools, while transparent voting and ranking harness collective judgement to prioritise the most promising ideas. Intelligent matching connects innovators with suitable collaborators and mentors, addressing a key barrier to idea progression.

In performance terms, elastic auto-scaling sustains low response times under the variable load characteristic of innovation communities, as the experimental results demonstrate. With respect to scalability, the cloud-native, stateless service design accommodates growth in participation without manual intervention or over-provisioning, and the use of managed data services and caching ensures responsive interaction. The web-based interface broadens accessibility, enabling participation from diverse contributors without specialised software.

VIII. LIMITATIONS

The present work has several limitations. The collaborator-matching recommendation depends on the richness and accuracy of user profiles, so its suggestions are less precise for new users with sparse profiles, a manifestation of the cold-start problem. Community evaluation through voting may be susceptible to popularity bias, potentially disadvantaging niche or unconventional ideas. The evaluation employed simulated load and a limited participant cohort over a bounded period, so broader, longer-term deployment may reveal additional behavioural dynamics. Finally, the platform is built upon a single cloud provider's services, and portability has not been validated.

IX. FUTURE ENHANCEMENTS

Future development will pursue several directions. Incorporating advanced machine-learning and natural-language-processing techniques would enhance collaborator matching and enable semantic analysis of idea content, including duplicate detection and trend identification. Refining the evaluation mechanism to mitigate popularity bias, for instance through weighted or expert-augmented scoring, would surface a broader range of promising ideas. Integrating real-time collaboration features, such as live co-editing and video discussion, would enrich interaction. The addition of analytics for tracking idea maturity and predicting success potential could guide investment decisions, and extending the platform toward a multi-cloud deployment would improve resilience and reduce lock-in.

X. CONCLUSION

This paper presented a cloud-based collaborative platform that manages the full lifecycle of startup ideas, from submission through community refinement, evaluation, and maturation. Built upon a Java Spring Boot backend deployed on elastic cloud infrastructure and a Node.js web client, the platform enables idea submission, threaded collaboration, community voting and ranking, and intelligent matching of ideas to relevant collaborators and mentors. Experimental evaluation under simulated load demonstrated low and gracefully degrading latency, a collaboration-efficiency score of 93.4%, and a recommendation-relevance score of 90.2%, consistently and substantially outperforming a single-server baseline across scalability, collaboration efficiency, recommendation relevance, and satisfaction. The principal contributions are a cloud-native architecture for collaborative idea management that unifies submission, evaluation, and matching, a community-driven evaluation and collaborator-matching mechanism that accelerates idea refinement, and an empirical demonstration of improved outcomes relative to conventional approaches. By uniting collective evaluation with intelligent collaboration and elastic infrastructure, the proposed platform offers a practical foundation for nurturing innovation at scale, with clear avenues toward semantic idea analysis, bias-aware evaluation, and real-time collaboration that promise to deepen its impact on the startup ecosystem.

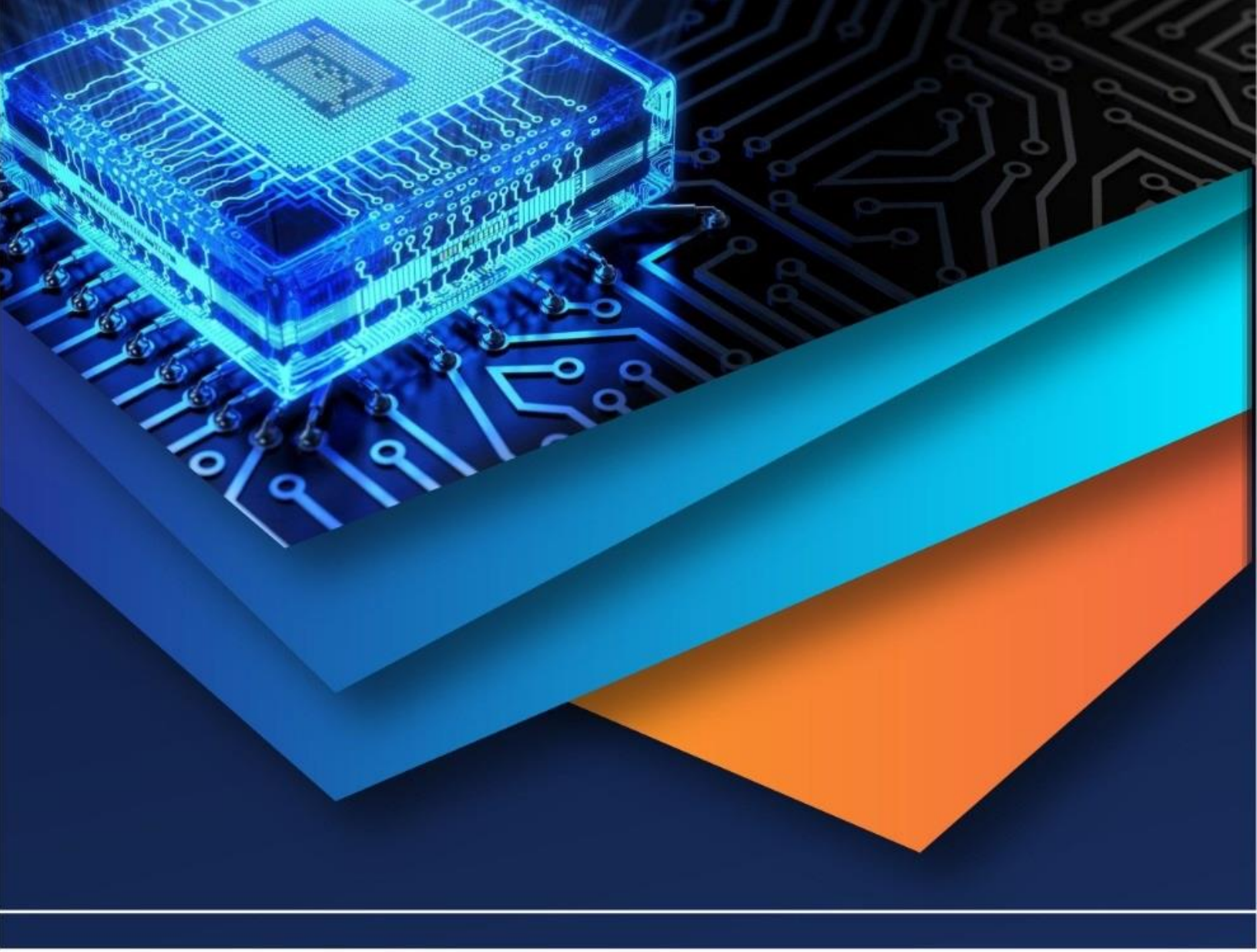
REFERENCES

[1] OECD, The Measurement of Scientific, Technological and Innovation Activities: Oslo Manual 2018, Paris: OECD Publishing, 2018.
[2] H. W. Chesbrough, Open Innovation: The New Imperative for Creating and Profiting from Technology, Boston, MA: Harvard Business School Press, 2003.
[3] J. Howe, "The rise of crowdsourcing," Wired Magazine, vol. 14, no. 6, pp. 1–4, 2006.
[4] L. B. Jeppesen and K. R. Lakhani, "Marginality and problem-solving effectiveness in broadcast search," Organization Science, vol. 21, no. 5, pp. 1016–1033, 2010.
[5] K. M. Bartol and A. Srivastava, "Encouraging knowledge sharing: the role of organizational reward systems," Journal of Leadership & Organizational Studies, vol. 9, no. 1, pp. 64–76, 2002.
[6] F. Ricci, L. Rokach, and B. Shapira, Recommender Systems Handbook, 3rd ed., New York: Springer, 2022.
[7] C. Baldwin and E. von Hippel, "Modeling a paradigm shift: from producer innovation to user and open collaborative innovation," Organization Science, vol. 22, no. 6, pp. 1399–1417, 2011.
[8] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the World-Wide Web," Communications of the ACM, vol. 54, no. 4, pp. 86–96, 2011.
[9] J. Bullinger, A. Neyer, M. Rass, and K. M. Möslein, "Community-based innovation contests: where competition meets cooperation," Creativity and Innovation Management, vol. 19, no. 3, pp. 290–303, 2010.
[10] P. R. Magnusson, "Exploring the contributions of involving ordinary users in ideation of technology-based services," Journal of Product Innovation Management, vol. 26, no. 5, pp. 578–593, 2009.
[11] O. Gassmann, E. Enkel, and H. Chesbrough, "The future of open innovation," R&D Management, vol. 40, no. 3, pp. 213–221, 2010.
[12] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," Knowledge-Based Systems, vol. 46, pp. 109–132, 2013.
[13] M. J. Brzozowski, T. Sandholm, and T. Hogg, "Effects of feedback and peer pressure on contributions to enterprise social media," in Proc. ACM Int. Conf. on Supporting Group Work, 2009, pp. 61–70.
[14] T. Lorido-Botran, J. Miguel-Alonso, and J. A. Lozano, "A review of auto-scaling techniques for elastic applications in cloud environments," Journal of Grid Computing, vol. 12, no. 4, pp. 559–592, 2014.
[15] M. Armbrust et al., "A view of cloud computing," Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.
[16] C. Walls, Spring Boot in Action, 2nd ed., Shelter Island, NY: Manning Publications, 2022.
[17] E. Estellés-Arolas and F. González-Ladrón-de-Guevara, "Towards an integrated crowdsourcing definition," Journal of Information Science, vol. 38, no. 2, pp. 189–200, 2012.

AUTHORS' BIOGRAPHIES

NILAM SAIKRISHNA received the B.Sc. degree from S.V.K.P. & Dr. K.S. Raju Arts and Science College, Penugonda, West Godavari, India, in 2024. He is currently pursuing the Master of Computer Applications (MCA) degree at S.V.K.P. & Dr. K.S. Raju Arts and Science College, Penugonda, West Godavari, India. Her academic interests include web designing,cloud computing, serverless architectures, cloud-native application development, financial technology systems, and software engineering. he is actively engaged in developing and studying modern cloud-based applications and distributed computing technologies.

K. LAKSHMANA REDDY Working as Associate Professor in S.V.K.P & Dr. K.S Raju Arts and Science College(A), Penugonda, West Godavari District, A.P. Master's Degree in Computer Applications from Andhra University 'C' level from DOEACC, New Delhi and MTech from Acharya Nagarjuna University, AP. He attended and presented papers in conferences and seminars. He has done online certifications in several courses from NPTEL. His areas of interest include Computer Networks, Network Security and Cryptography, Formal Languages and Automata Theory and Object-Oriented programming languages.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY