

THINKNOVA – An AI Powered College Startup Incubation Platform

M.D. Shalini¹, L.G.S. Srinidhi², Dr. A. Askarunisa³, A. Alagar⁴

^{1, 2, 3} Dept of CSE

⁴ Assist prof, Dept of CSE

^{1, 2, 3, 4} K.L.N.College of Engineering, Sivagangai, India

Abstract- *Student entrepreneurs often lack structured idea validation, expert mentorship, and access to incubation resources. This paper presents ThinkNova, an AI-powered startup incubation and mentorship platform that guides campus entrepreneurs from idea submission to incubation approval. The system offers role-based portals for Students, Alumni Founders, and Incubation Administrators. An AI engine provides idea validation, business model generation, MVP roadmap planning, competitor analysis, and mentor matching. ThinkNova is built on a layered architecture with a React client, Spring Boot services, JWT-secured authentication, and a PostgreSQL database deployed on AWS. By combining automated analysis with human mentorship, ThinkNova shortens the path from idea to execution and gives students a transparent incubation workflow.*

Keywords- Artificial Intelligence, Idea Validation, Mentor Matching, Spring Boot, Startup Incubation

I. INTRODUCTION

Engineering colleges increasingly encourage entrepreneurship, and many students begin with promising ideas. Turning an idea into a working venture, however, needs validated market demand, a sound business model, a realistic development plan, and guidance from people who have built companies before. Business incubators exist to provide this support [1], [2], but on campus the support is often informal, scattered across faculty and personal contacts, and hard to track.

Campus incubation cells face practical problems. Manual review of every submitted idea is slow, feedback varies between reviewers, and students have little visibility into where their idea stands. Alumni founders are willing to mentor but have no structured way to discover relevant student teams. Established frameworks such as the lean startup method [3] and the business model canvas [4] tell founders what to examine, but applying them to each idea still requires time and expertise.

Large language models built on the Transformer architecture [5], such as GPT-3 [6] and Gemini [7], make it

feasible to automate a first-level analysis of an idea. They can summarise a problem statement, draft a business model, suggest an MVP scope, and compare an idea with existing solutions. Such output does not replace human judgement, but it gives students fast and consistent feedback and gives mentors a better starting point. Mentor selection can in turn be treated as a recommendation problem [8], [9].

This paper presents ThinkNova, an AI-powered college startup incubation platform. Its main contributions are: (i) an end-to-end digital workflow from registration to incubation approval; (ii) three role-based portals for Students, Alumni Founders, and Incubation Administrators; (iii) an AI evaluation engine that scores an idea and produces a structured report; (iv) explainable mentor recommendations with a mentorship request workflow; and (v) a layered, JWT-secured architecture built with React, Spring Boot, and PostgreSQL on AWS.

The rest of the paper covers the methodology (Section II), the drawbacks of the existing system (III), the proposed solution (IV), performance enhancement measures (V), and results and discussion (VI), and ends with the conclusion (VII).

II. METHODOLOGY

ThinkNova was developed iteratively. Each iteration added one working vertical slice (database, service, and interface) and was tested before the next was started. The work followed five stages.

A. Requirement Analysis

The needs of the three user groups were identified first. Students need a way to submit ideas and receive feedback; alumni founders need a way to find relevant teams and offer guidance on their own schedule; administrators need a single place to review ideas, AI reports, and mentor feedback. These needs led to the role-based design of the platform.

B. System Design

A layered design was chosen to separate the client, security, application, intelligence, and data concerns (Section IV). Functionality is exposed through REST endpoints [10], and the relational schema holds users, ideas, AI reports, mentor profiles, mentorship requests, availability slots, and scheduled sessions.

C. AI Evaluation Design

A student submits an idea draft through a structured form with the title, a short concept description (limited to 512 characters), category, development stage, visibility, target industry, target audience, problem statement, and proposed solution. These fields form the input to the evaluation engine. The engine returns three scores out of 100 (market demand, innovation and moat, and feasibility), an overall risk level, an executive incubation summary, and detailed analysis grouped into five areas: problem and market, technology and operations, business model, MVP roadmap, and risks and assumptions.

D. Mentor Matching Design

Alumni founders register with their domain, skills, and experience. For a given idea, the platform ranks mentors by domain alignment, technical competencies, and the guidance the student requests. Each recommendation shows a match percentage and a short "why recommended" list that covers domain alignment, skill match, coverage of gaps identified in the AI report, and founder experience. The student can then send a mentorship request.

E. Security and Testing

Authentication uses Spring Security with JSON Web Tokens (JWT) [11], and the role carried in the token controls access to each portal. The prototype was tested functionally using separate student and alumni accounts and a sample idea that was taken through the full journey, from registration to an accepted mentorship request (Section VI).

III. DRAWBACKS OF EXISTING SYSTEM

Incubation support on most engineering campuses is still informal and largely manual. The main drawbacks of the existing system are as follows.

1. Informal and scattered support: guidance depends on faculty and personal contacts instead of one organised platform, so it is hard to track and easy to lose.
2. Slow manual idea review: every submitted idea is reviewed by hand, which delays feedback and limits the number of ideas an incubation cell can handle.
3. Inconsistent feedback: different reviewers apply different criteria, so similar ideas can receive different responses.
4. No visibility for students: students have little information about where their idea stands in the review process.
5. No structured mentor discovery: alumni founders are willing to mentor but have no way to find relevant student teams, and students have no systematic way to find a suitable mentor.
6. Dependence on expertise: frameworks such as the lean startup method [3] and the business model canvas [4] show what to examine, but applying them to each idea takes time and expertise that students and reviewers often lack.
7. No lasting record: feedback and mentor interactions are not stored in one place, so the history of an idea cannot be reviewed later.

IV. PROPOSED SOLUTION

ThinkNova addresses these drawbacks with a single web platform that guides a campus idea from registration to incubation approval. It combines automated first-level analysis with human mentorship, and its main features are as follows.

1. Role-based portals: separate portals for Students, Alumni Founders, and Incubation Administrators, with access controlled by the role carried in a JWT [11].
2. Structured idea submission: one form collects the title, concept, category, stage, target industry, target audience, problem statement, and proposed solution, which replaces informal and scattered intake.
3. AI evaluation: the engine scores each idea out of 100 for market demand, innovation and moat, and feasibility, and returns a risk level, an executive summary, and a detailed analysis. Every idea is judged on the same criteria and receives feedback quickly.
4. Explainable mentor matching: mentors are ranked by domain alignment, skills, and the guidance requested, and each recommendation shows a match percentage and the reasons for it. The student then sends a mentorship request that carries the stored evaluation snapshot.
5. Transparent workflow: the journey runs through registration, idea submission, AI processing, report review, mentor connection, progress tracking, and

administrator review. Each step is recorded, and if an idea is not approved, the feedback returns to the student, who can refine and resubmit.

- Layered and maintainable architecture: a React client, Spring Security with JWT, Spring Boot services [10], and a separate AI engine, with PostgreSQL for data and AWS for deployment. Because the AI engine is a distinct service, the provider, including the Gemini API [7], can be changed without affecting the rest of the system.

ThinkNova presents the AI output as decision support. The final decision stays with human mentors and administrators.

V. PERFORMANCE ENHANCEMENT

The following design measures were adopted to keep the platform responsive and inexpensive to run.

- Evaluate once, reuse everywhere: the AI report for an idea is generated once and stored. The report tabs and the dashboards then read it from the database instead of calling the AI service again, which reduces response time and API usage.
- Evaluation snapshot in requests: a mentorship request carries the idea's scores, risk level, and summary (Fig. 5), so a mentor sees the context immediately without extra computation or navigation.
- Stateless authentication: JWT removes the need for server-side session storage, so the services can be scaled horizontally without sharing session state.
- Bounded input: limiting the concept description to 512 characters keeps the evaluation input small and consistent.
- Separation of the AI engine: because the engine is a distinct service, the model can be swapped for a faster or cheaper one with little change to the application.
- Free-tier model usage: the AI features are designed to run within free-tier model quotas, which keeps the platform viable for a college without a dedicated budget.
- Filtered views: mentors can filter requests by status (All, Pending, Accepted, Rejected, Cancelled), and students can filter recommended mentors by the guidance they need, which keeps lists short and relevant.

A quantitative benchmark of response time and throughput has not been carried out yet and is planned as future work.

VI. RESULTS AND DISCUSSION

A. Functional Walkthrough

The prototype was evaluated by following one sample idea, "finora" (a finance management idea in the FinTech category), through the platform with a student account and an alumni account. Registration and login are shown in Fig. 1. The student creates the idea draft in the form of Fig. 2, which collects the fields described in Section II. On submission, the evaluation engine generates the report of Fig. 3, and the platform then lists recommended alumni mentors (Fig. 4).

Fig. 1. (a) Registration with role selection and (b) login screens

Fig. 2. Idea draft form used for submission

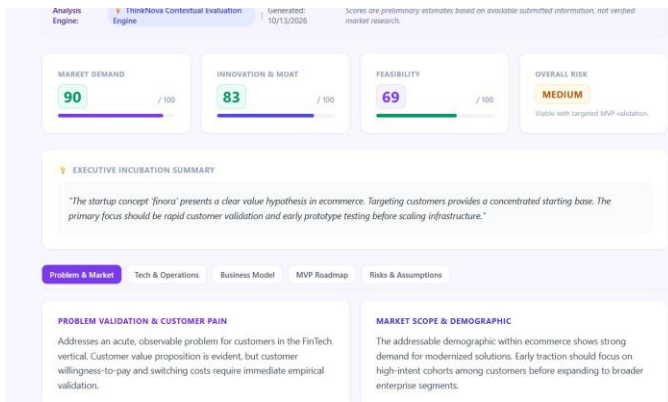


Fig. 3. AI evaluation report with scores, risk level, executive summary, and analysis tabs

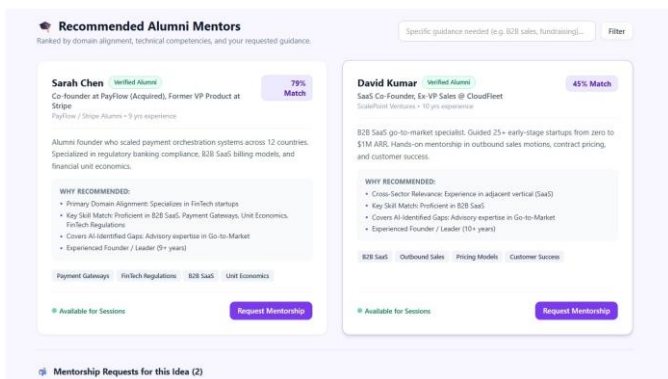


Fig. 4. Recommended alumni mentors with match percentage and reasons

B. Observed Results

Registration and access. The sign-up screen lets a user choose the Student (incubatee) or Alumni (founder / mentor) role, and the same role then determines which portal opens after login (Fig. 1).

AI evaluation. For the sample idea, the engine returned a market demand score of 90, an innovation and moat score of 83, a feasibility score of 69, and a medium overall risk (Fig. 3). The report states that the scores are preliminary estimates based on the submitted information and are not verified market research. The lowest score, feasibility, is consistent with a risk level of medium, and the summary advises rapid customer validation and prototype testing before scaling.

Mentor recommendation. Two mentors were recommended. A FinTech payments founder was ranked first with a 79% match, and a B2B SaaS go-to-market specialist was ranked second with 45% (Fig. 4). The ranking agrees with the idea's FinTech domain, since the first mentor matches the primary domain while the second matches only an adjacent

sector. The "why recommended" lists let the student see the reason for each ranking.

Mentorship workflow. The request reached the mentor with the idea's category, stage, requested guidance, and the stored AI evaluation snapshot. The mentor accepted it and sent a reply to the student (Fig. 5). The mentor portal also provides tabs for availability slots and scheduled sessions, so an accepted request can be followed by a booked session.

C. Discussion

The results show that the main student-to-mentor path works end to end: a student can register, submit an idea, receive a structured evaluation, find a suitable mentor, and obtain a mentor's response within one platform. Compared with informal mentoring, the process is faster to start, uses the same criteria for every idea, and leaves a visible record of each step.

Several limitations remain. The scores are estimates produced from the submitted text alone and are not validated against real market data. In the sample report, some wording is fairly generic and the idea's category (FinTech) and industry (ecommerce) are described together without reconciling them, which shows that the quality of the analysis depends on the quality of the input and on the prompt design. The mentor profiles used in testing were sample profiles, so the match percentages demonstrate the ranking behaviour and not real-world mentoring outcomes. The evaluation was also limited to a single sample idea and has no user study yet. For these reasons ThinkNova presents the AI output as decision support, and the final decision stays with human mentors and administrators.

VII. CONCLUSION

This paper presented ThinkNova, an AI-powered startup incubation and mentorship platform for college entrepreneurs. The platform combines role-based portals, a seven-step workflow, an AI evaluation engine, explainable mentor recommendations, and a mentorship request process on a layered React, Spring Boot, JWT, and PostgreSQL architecture. A walkthrough with a sample idea showed that a student can move from registration to an accepted mentorship request inside the platform, with scores, a risk level, and a ranked list of mentors produced along the way.

Future work includes completing the administrator review and incubation approval workflow, validating the evaluation scores against real market data, conducting a user study with students, mentors, and administrators, and

measuring response time and throughput. Planned extensions include a funding portal, investor connections, patent guidance, a startup marketplace, live chat, and a success dashboard that tracks incubated ventures.

VIII. ACKNOWLEDGMENT

The authors thank the Department of Computer Science and Engineering, K.L.N. College of Engineering, for their guidance and support during this work.

REFERENCES

- [1] P. Hausberg and S. Korreck, "Business incubators and accelerators: A co-citation analysis-based, systematic literature review," *J. Technol. Transf.*, vol. 45, pp. 151–176, 2020.
- [2] A. Bergek and C. Norrman, "Incubator best practice: A framework," *Technovation*, vol. 28, no. 1–2, pp. 20–28, 2008.
- [3] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. New York, NY, USA: Crown Business, 2011.
- [4] A. Osterwalder and Y. Pigneur, *Business Model Generation*. Hoboken, NJ, USA: Wiley, 2010.
- [5] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 5998–6008.
- [6] T. B. Brown et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 1877–1901.
- [7] Gemini Team, Google, "Gemini: A family of highly capable multimodal models," *arXiv:2312.11805*, 2023.
- [8] F. Ricci, L. Rokach, and B. Shapira, Eds., *Recommender Systems Handbook*, 2nd ed. New York, NY, USA: Springer, 2015.
- [9] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. EMNLP-IJCNLP*, 2019, pp. 3982–3992.
- [10] R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, Univ. California, Irvine, CA, USA, 2000.