

Vibe Exploration: Human–AI Collaboration for Problem-Space Exploration in Software Startups

Daniel Planötscher

Free University of Bolzano-Bozen
Bolzano, Italy

Sebastiano Druda

Free University of Bolzano-Bozen
Bolzano, Italy

Abstract

Context: Many software startups fail by building solutions before understanding real user needs. Meanwhile, AI in innovation is used mainly for solution development, where human–AI collaboration accelerates prototyping and delivery. *Objective:* This study investigates how human–AI collaborative approaches can support problem-space exploration in software startups. *Method:* We report a case study of an early-stage software venture that systematically used human–AI collaboration for problem-space exploration. *Findings:* We reconstruct the observed exploration process and identify key limitations and potentials. Limitations include overreliance on secondary data, confirmation bias, and premature closure of the problem-space. Potentials include faster synthesis of large data and rapid prototype development for learning. Based on these insights, we propose a refined, hypothesis-driven workflow grounded in iterative Build–Measure–Learn cycles. *Conclusion:* Human–AI collaboration can strengthen problem-space exploration when combined with open-ended primary research and continuous user interaction.

CCS Concepts

• **Human-centered computing** → **Empirical studies in HCI**; • **Software and its engineering** → *Software creation and management*.

Keywords

human–AI collaboration, generative AI, problem-space exploration, software startups, problem–solution fit, case study

ACM Reference Format:

Daniel Planötscher and Sebastiano Druda. 2026. Vibe Exploration: Human–AI Collaboration for Problem-Space Exploration in Software Startups. In *Proceedings of the 30th International Conference on Evaluation and Assessment in Software Engineering (EASE 2026)*, June 9–12, 2026, Glasgow, Scotland, United Kingdom. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 Introduction

Innovation is a core driver of economic growth and technological advancement [10], and startups are among its primary vehicles, translating novel ideas into new products and services. Yet most

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
EASE 2026, Glasgow, Scotland, United Kingdom

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

startups fail [1], often because they do not address genuine market needs¹ and end up building solutions to problems that do not exist.

In recent years, AI² has been increasingly integrated into innovation processes [13]. Although it shows promise for addressing this challenge of understanding genuine market needs, for example by synthesising customer feedback and clarifying demand signals, its use in innovation has so far focused primarily on development activities. In software startups, for instance, AI has proven especially useful for accelerating prototyping [2], particularly through human–AI collaborative approaches such as vibe coding [7], which enable developers to build applications through low-friction, conversational interaction at unprecedented speed.

This creates a significant gap. Although solution development is accelerating, building products that customers actually want requires working across two distinct spaces [11]: the *problem-space*, focused on understanding users, their needs, and the contexts in which those needs arise, and the *solution space*, focused on designing and building a product that addresses them. While GenAI is increasingly applied to the solution space, its potential to support the problem-space, arguably the more critical of the two, remains largely unexplored. This raises a key question:

RQ: How can human–AI collaboration support problem-space exploration in software startups?

To answer this, we present a case study of a young software venture actively using AI tools to explore the problem-space. We examine how the team conducts this exploration, where AI enhances the process, and where it risks constraining problem discovery. From these findings, we derive a high-level approach and outline directions for future research.

The rest of the paper is structured as follows: Section 2 presents the necessary background and related work in the field of AI for user research, Section 3 details the research method, Section 4 highlights the results, Section 5 discusses the findings, Section 6 outlines threats to validity, and Section 7 concludes the paper.

2 Background and Related Work

2.1 Problem–Solution Fit

Problem–solution fit is the first of the three key stages of the startup journey according to Maurya’s *Running Lean* [9]. It marks the moment when a real problem worth solving is uncovered and a solution is identified that addresses it, often described as the point “where the rubber meets the road” [9].

Reaching problem–solution fit is particularly critical for startups. Since customers do not buy products per se but the promise of something better [9], adoption depends heavily on a value proposition

¹<https://www.cbinsights.com/research/report/startup-failure-reasons-top/>

²In this paper, we use *AI* and *GenAI* interchangeably to refer to contemporary generative AI systems and tools.

that addresses real and meaningful customer needs. Without this alignment, startups risk building solutions nobody wants, which remains one of the leading causes of startup failure.

Several techniques offer guidance for navigating this process. Design thinking, particularly the Double Diamond framework³, structures this ambiguity through alternating divergence and convergence: *discover* the context and user needs, *define* a focused problem statement, *develop* candidate solutions, and *deliver* a validated one. Similarly, Steve Blank’s Customer Development [3] argues that business plans should be validated through direct customer interaction by surfacing assumptions and testing hypotheses.

2.2 Problem Exploration in Practice

Deeply understanding customer problems is especially challenging when aiming to achieve problem–solution fit, because users often do not know what they truly need or cannot clearly articulate it. Uncovering genuine needs therefore requires direct, open-minded engagement with people in their context [3], for example through semi-structured interviews to surface motivations and uncover underlying needs [11], or through observation in natural environments to capture real behaviours.

Doing this well requires access to real users, the ability to interpret observations correctly, and the discipline to apply research methods rigorously. For early-stage startups, where time is a scarce resource, these requirements are particularly hard to meet, often resulting in solutions that fail to address any genuine need.

2.3 GenAI for Problem Exploration

To address these concerns, a few studies have begun exploring how GenAI might support understanding true user needs. A prominent line of work examines AI-generated personas [4], including interactive personas [5], and their use for eliciting early feedback without direct access to end users [15].

While early results are promising, persona-based approaches cover only a narrow part of problem-space exploration. As noted above, effective problem exploration requires deeper research and direct engagement with users, which synthetic personas cannot fully replace. Moreover, how humans and AI can collaborate in this process remains an open question.

2.4 Human–AI Collab in Solution Development

In contrast to the limited attention given to the problem-space, human–AI collaboration has advanced considerably on the solution side. Recent research shows how generative AI can support software development across the full lifecycle, from defining specifications to designing architectures to writing and testing code [16]. Agentic coding tools such as GitHub Copilot are now widely adopted⁴, and techniques like spec-driven development⁵ and vibe coding continue to accelerate delivery [7] through direct collaboration between humans and AI, combining AI’s speed in generating software code with human judgment for contextual relevance and quality control.

³<https://www.designcouncil.org.uk/our-resources/the-double-diamond/>

⁴<https://survey.stackoverflow.co/2025/ai>

⁵<https://github.blog/ai-and-ml/generative-ai/spec-driven-development-with-ai-get-started-with-a-new-open-source-toolkit/>

While human–AI collaboration is enhancing solution development, its role in the problem-space remains largely unexplored. Startups can now build faster than ever, yet the fundamental challenge of determining whether they are building the right thing remains largely unaddressed by current AI tooling. This research therefore investigates how human–AI collaborative approaches, comparable to those now established in the solution space, can be applied to problem-space exploration.

3 Research Method

To explore this gap, this research adopts a single-case study design. Case study is a common method in software engineering research [14], which is typically used to answer “why” and “how” questions and where a research phenomenon under the study and its context are difficult to separate [8]. A single-case study approach is specifically used to study a rare and concrete phenomenon in depth [8].

3.1 Case Selection

We purposefully selected a case that provides an early example of human–AI collaborative problem-space exploration in software startups, directly aligning with our study objectives. The selected case is a young software venture that developed a systematic approach for exploring the problem-space of their startup idea, largely relying on human–AI collaboration with currently available tools. The second author is a co-founder of this venture, which further motivated the case selection and provided privileged access to process artefacts and participants.

3.2 Data Collection

We collected data from multiple sources. We conducted both a 1.5-hour interview with one co-founder and a longer 2.5-hour participant observation session with both founders, during which we examined how they used generative AI to explore the problem-space and asked follow-up questions. In addition, we examined documents and artefacts produced by the founders, including product requirement documents and other materials developed during problem–solution fit, as well as their conversations with LLM chatbots.

3.3 Data Analysis

The first author, who was not part of the startup team, thematically analysed the interview and observation transcripts to map the team’s problem-exploration process. The findings were then triangulated with observed behaviour during the session to compare perceived and actual uses of AI. Next, the startup’s approach was assessed against problem–solution fit best practices to identify the potential and limitations of AI use in this phase. Finally, the second author reviewed the findings for factual accuracy but did not contribute to interpretation.

4 Results

4.1 Case Description

The selected case is an early-stage software startup based in Italy, founded by two developers with software-development backgrounds.

One founder works professionally as a software developer, while the other is completing a degree in business informatics, bringing both technical and business-oriented perspectives to the venture. The startup operates informally, with no external funding or dedicated working space.

In the past, the founders have explored various ideas for software products during their short history, allowing the systematic problem-exploration approach to emerge and be refined across iterations. To date, they have focused on three products spanning different contexts and targeting different user groups:

- (1) **Product A:** A financial and fiscal document management platform for independent professionals and micro-businesses in Italy, originating from the founders' own needs and subsequently repurposed as a commercial SaaS product.
- (2) **Product B:** A workforce time-tracking application targeting small businesses, developed as a SaaS product.
- (3) **Product C:** A vehicle maintenance tracking application with integrated AI capabilities, targeting automotive enthusiasts who currently rely on informal record-keeping methods. The concept was at an ideation stage, with no prior validation at the time of observation.

Given their circumstances, the team operates under significant time constraints and they primarily work during late evening hours on their projects.

4.2 Human-AI Collab in Problem Exploration

Drawing on the observational study and triangulating it with the earlier interview in which one founder outlined the workflow, we derived the following high-level process for human-AI collaboration in problem exploration:

- (1) **Identify assumptions:** The process starts with the two founders collaboratively creating a Markdown file that captures all of their ideas and assumptions about the project, without any help of AI. They do this at the very beginning to stay focused and avoid AI biasing their initial framing. The final file typically includes the problems their idea aims to solve and how much they believe these problems matter. It may also include potential customer segments and general information about them. One founder describes this: *"We always start from our own assumptions, our own idea of what we thought is okay to build."* After the Markdown file is developed, they ask GenAI to reformulate and restructure the document into a list of testable hypotheses, also improving terminology used.
- (2) **AI-augmented research:** Once the assumptions document is complete, the team collaborates with AI to validate their hypotheses. Specifically, they provide their final Markdown file to Perplexity⁶ and run a deep research query, asking the model to gather online evidence to assess each assumption. In this step, the AI is explicitly prompted to challenge assumptions rather than confirm them and to justify each assessment. One founder described the AI's role as follows:

"The AI manages to analyse and eventually criticise the weaknesses and vulnerabilities of the idea, rather than coming up with its own idea."

- (3) **Judgment:** Finally, the founders ask the AI to judge whether the project is worthwhile to continue and examine the AI's argumentation to develop their own decision.

From here, the approach diverges across projects and is guided by the AI's assessment:

- If the AI supports the idea, the team drafts a product requirements document (again with the help of GenAI), builds a prototype using AI-driven development, and explores it through build-measure-learn cycles.
- When the initial AI response is inconclusive or raises new questions, the founders refine their problem framing using the insights gained and repeat the analysis to deepen their understanding of specific ideas or problems. They describe this process as *brainstorming with AI*, in which they iteratively refine both their problem understanding and its underlying assumptions based on AI input, treating each response as an opportunity to refine their mental framing of the problem before re-engaging with the model. This process is further enabled by Perplexity's Spaces feature, which provides a shared workspace that preserves context across sessions.
- When asked what happens if the AI advises against an idea, one founder stated that this had not occurred in practice; instead, the AI mostly suggested pivoting, for instance by exploring specific niches within the proposed idea space that it considered less targeted.

4.3 Limitations

While the approach above shows an interesting first look into how problem-space exploration might be performed in collaboration with AI, that need to be addressed:

4.3.1 Confirmation over Exploration. The most concerning limitation is that the process is structured for confirmation rather than open problem-space exploration. Instead of using AI to surface unknowns and challenge assumptions, the founders first define their idea in a structured form and then ask AI to assess whether that framing is plausible. As one founder put it, they *"ask the LLM basically if there is the need or if there are people that might be interested."* In other words, the central goal shifted from uncovering true customer needs to checking whether their initial idea is correct.

Notably, the founders reported trying open-ended AI ideation but rejecting it because it produced too much divergence: *"we prefer to start with the concrete assumption from our own experience and idea... when you ask for assumptions with a not well-structured prompt it comes out with information that are from different fields and different situations that most of the time are totally out of context."* As a result, the approach is deliberately designed to narrow the problem space based on a pre-formed idea, not to open it up for new insights.

This also explains why even explicit requests for critique failed to reopen exploration. During the ethnographic session, for instance, one founder instructed AI to criticise their thinking so that it would not be *"biased [...] on what we tell it."* Yet AI rarely challenged the

⁶<https://www.perplexity.ai/>

underlying idea and instead suggested nearby pivots or incremental refinements. This outcome was likely a direct consequence of process design: by providing AI with a structured framing upfront, the founders had already set the boundaries within which AI could respond, making it difficult for the model to move beyond that frame.

4.3.2 Overreliance on Secondary Data. This bias toward confirmation is further reinforced by an overreliance on secondary data in place of primary research. Because the founders typically postpone direct conversations with customers until later stages and rely mainly on information available online, they are much less likely to encounter genuinely surprising insights, as is often the case in primary research. As a result, genuinely new observations rarely surface, few hypotheses are fundamentally contradicted, and reframing remains anchored to what is already available online.

According to one of the founders, secondary data is viewed as a viable substitute for direct customer interaction in the early stages: *“we prefer to talk with the niche so we talk with the niche with the articles, the information... all the information that you found around those applications that you want to build, and we come up to our own idea.”* The founders also argue that practical constraints reinforce this reliance on secondary data. When asked *“why don’t you ever just go to people and talk to them about this?”* the founder responded: *“we are both employees, I’m also a student... we discuss with LLM because it is easier — people are not available at night.”*

Therefore, Steve Blank’s “get out of the building” principle [3], which emphasizes primary research as a necessary component, should not be neglected even as new tools emerge. Only by talking to people and understanding their actual needs are we able to uncover problems truly worth solving.

4.3.3 Building as a Response. As a result of the limitations above, the problem-space is typically closed down prematurely, leading to them building solutions even before a genuine customer need is discovered. While building may serve as a learning tool that allows for quick validation, it only does so when the underlying problem is meaningful and well understood. In the context of these founders, the logic runs closer to “build it and they will come”, as one founder argues: *“if you build something that you want and you are still part of a niche, there will be people that want it. It is a product that attracts people; not the people that attract the product.”*

While the tendency to build before validating has always been a known pitfall, agentic coding tools appear to amplify it. As one founder put it, *“an MVP instead of taking six or eight weeks now takes maybe one week.”* This removes the friction that once separated an unvalidated idea from a functional product and, in doing so, can prematurely lock founders’ mental models into a specific solution space before the problem is genuinely understood, making it increasingly difficult to fundamentally revise the core idea or pivot toward a better-aligned solution.

4.4 Potentials

While these limitations are concerning, we can also identify several possibilities of AI in exploring the problem-space.

4.4.1 Rapid Competitor Discovery. The case shows that AI can be used to rapidly scan the market for competitors through capabilities

such as deep research. By describing their idea in detail, as the founders did with their Markdown file, AI was able to search the internet for similar products and uncover potential competitors early. As a result, the founders were able to quickly understand the market context of their idea. As one founder noted: *“I wouldn’t have found it, probably, with my own search.”*

Beyond this, AI can also understand how similar products are positioned on the market and where potential market gaps are. With this, AI helped them position their products better in the market, highlighting for which kind of target customer no specific product was yet available. After running a competitor analysis through Perplexity, the founders noted that the AI output identified specific underserved segments, such as motorcycle owners and users who own multiple vehicles, which were not present in any existing solution.

4.4.2 Realistic Personas. Another interesting use case of AI observed during this study was the development of realistic customer personas. Using large chunks of publicly available information, such as online conversations on platforms like Reddit, AI was able to quickly surface the opinions and unmet needs of different people, and propose distinct personas with their respective user journeys. This provided an initial overview of who potential target customers might be, without requiring weeks of prior fieldwork.

In addition to identifying personas, this approach helped the founders understand the specific workflows customers perform regularly and the friction points within them. By creating representations of their target customers that were both realistic and accessible, it became easier to identify concrete problems those customers might face, allowing them to refine their problem-space understanding.

In its current form, however, the technique remains limited because the generated representations rely only on secondary data and are therefore less likely to surface entirely new insights. To address this issue, future work should explore applying the same technique to primary research outputs, such as interview transcripts or field notes from observational studies, alongside secondary sources.

4.4.3 Rapid Prototype Development. Although early solution-space development may have negative consequences, building tangible solutions can become a powerful learning tool once the problem is sufficiently understood. With the introduction of agentic coding tools, this process has become faster than ever, compressing the time between an idea and a working prototype. As a result, known customer problems can be understood more deeply, refining both problem-space and solution-space understanding through faster iteration.

5 Discussion

The findings above demonstrate both notable limitations and significant potential for human–AI collaboration in problem exploration. While using AI in problem exploration might lead to a bias toward confirming assumptions, overreliance on secondary data, and a failure to sufficiently open up the problem-space, it also shows clear strengths: the ability to synthesise large volumes of information rapidly and to accelerate early prototyping. This suggests that AI

is not inherently unsuited to this process, but rather that the way it is applied requires refinement.

We use these observations as the basis for refining the technique into a process more closely aligned with well-established startup practices. Specifically, we propose a highly iterative, hypothesis-driven approach that builds on methods such as the Build-Measure-Learn (BML) loop from the Lean Startup [12] and continuous experimentation techniques such as the RIGHT model [6].

As a result, we propose the following approach for human-AI collaborative problem exploration:

- (1) **Build:** Surface underlying assumptions and formulate explicit hypotheses about the problem-space, as the two founders are already doing.
- (2) **Measure:** Conduct AI-collaborative research in a more open-ended way to test and expand these hypotheses, drawing on both secondary data and primary research methods such as interviews. This mitigates overreliance on secondary sources and keeps the process open to new insights.
- (3) **Learn:** Synthesize findings in collaboration with AI to refine understanding of the problem-space and derive new directions for future learning. These directions then inform a new round of hypotheses, starting the next cycle.

The process is illustrated in Figure 1. Reframing the process through this lens preserves the strengths of AI in synthesising large volumes of information, while guarding against the closed-mindedness that characterised the observed approach.

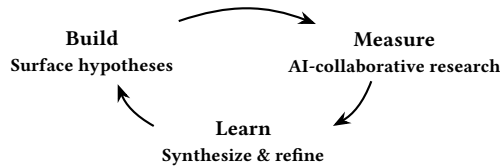


Figure 1: Build-Measure-Learn Loop [12] Adapted for Human-AI Collaborative Problem Exploration.

Once the problem-space is sufficiently understood and a meaningful problem has been identified, tangible prototypes can be used to further explore the problem-space and deepen understanding of the problem the team is trying to solve. At this stage, the traditional Build-Measure-Learn loop takes over, continuing to shape problem-space understanding through iterative learning.

6 Threats to Validity

As this is a single-case study, the findings cannot be generalized. The focus is therefore on documenting a case where human-AI collaboration is already used for problem exploration, and on deriving lessons for how such a process might be refined.

A related concern is self-report bias, given that both participants are also the founders of the venture. To address this concern, an ethnographic observation component was conducted alongside the interview, even though observation itself may have influenced behavior. Reassuringly, the observed process aligned closely with the initial interview. Since the projects are at early stages, no claims are made about the effectiveness of the approach; the goal is to

understand how it is currently practiced in one case, and to surface potential improvements.

Finally, although the second author is a member of the studied team, their involvement was limited to verifying factual accuracy. They played no role in interpreting the data or deriving conclusions.

7 Conclusions

In conclusion, this paper shows that human-AI collaboration can support problem-space exploration in software projects, but it must be done with care. While it accelerates the synthesis of large amounts of data and enables faster prototyping, it might also narrow the problem-space too early and reinforce assumptions when primary research is lacking. To counteract these issues, this paper proposes refining the process toward hypothesis-driven development approaches, where the problem-space is explored through more open primary research while still leveraging AI to capture information and learn from insights.

This research has implications for both practitioners and researchers. For practitioners, we present an approach to problem-space exploration that leverages human-AI collaboration to synthesise large volumes of data while mitigating the identified limitations. For researchers, we highlight concrete avenues for future work: refining the proposed model, developing human-AI collaborative tooling that supports open-ended exploration, and evaluating the approach across a broader range of ventures and contexts.

References

- [1] Muhammad Naeem Akhtar and Kamarulzaman Ab Aziz. 2025. Understanding the dynamics of startup success: A systematic literature review. *PaperASIA* 41, 1b (2025), 283–291.
- [2] Volker Bilgram and Felix Laarmann. 2023. Accelerating innovation with generative AI: AI-augmented digital prototyping and innovation methods. *IEEE engineering management review* 51, 2 (2023), 18–25.
- [3] Steve Blank. 2020. *The Four Steps to the Epiphany: Successful Strategies for Products That Win*. Wiley.
- [4] Stefano De Paoli. 2025. User personas, ideation and large language models: A post-hoc study. *International Journal of Human-Computer Studies* (2025), 103690.
- [5] Paluck Deep, Monica Bharadhidasan, and A Baki Kocaballi. 2025. “She was useful, but a bit too optimistic”: Augmenting Design with Interactive Virtual Personas. *International Journal of Human-Computer Studies* (2025), 103646.
- [6] Fabian Fagerholm, Alejandro Sanchez Guinea, Hanna Mäenpää, and Jürgen Münch. 2017. The RIGHT model for continuous experimentation. *Journal of Systems and Software* 123 (2017), 292–305.
- [7] Ahmed Fawzy, Amjed Tahir, and Kelly Blincoe. 2025. Vibe Coding in Practice: Motivations, Challenges, and a Future Outlook—a Grey Literature Review. *arXiv preprint arXiv:2510.00328* (2025).
- [8] Yin K Robert. 2018. Case study research and applications design and methods. Library of Congress Cataloging-in-Publication Data.
- [9] Ash Maurya. 2012. *Running lean : iterate from plan A to a plan that works*. O’Reilly, Sebastopol, CA.
- [10] Maha Mohamed Alsebai Mohamed, Pingfeng Liu, and Guihua Nie. 2022. Causality between technological innovation and economic growth: Evidence from the economies of developing countries. *Sustainability* 14, 6 (2022), 3586.
- [11] Dan Olsen. 2015. *The lean product playbook: How to innovate with minimum viable products and rapid customer feedback*. John Wiley & Sons.
- [12] Eric Ries. 2011. *The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business, New York.
- [13] Deborah L Roberts and Marina Candi. 2024. Artificial intelligence and innovation management: Charting the evolving landscape. *Technovation* 136 (2024), 103081.
- [14] Per Rumeson, Martin Host, Austen Rainer, and Bjorn Regnell. 2012. *Case study research in software engineering: Guidelines and examples*. John Wiley & Sons.
- [15] Mario Simaremare and Henry Edison. 2025. Active Personas for Synthetic User Feedback: A Design Science Study. In *International Conference on Software Business*. Springer, 251–266.
- [16] Valerio Terragni, Annie Vella, Partha Roop, and Kelly Blincoe. 2025. The future of ai-driven software engineering. *ACM Transactions on Software Engineering and Methodology* 34, 5 (2025), 1–20.