

# AI and Teaching - The Brave New World

Steve Blank (Stanford University)

**KEYWORDS:** Teaching Methods, Lean Startup, Lean startup method.

AI is accelerating product development in Lean LaunchPad classes -- but students are overwhelmed and customers are worried about the future.

This is the 16th year we've been teaching the [Stanford Lean LaunchPad](https://leanlaunchpad.stanford.edu/) class. This year, from the first hour of the first class, we realized we were seeing something extraordinary happen. It was both the end and beginning of a new era.

Teams showed up to the first day of class with MVPs (Minimum Viable Products) looking like finished products that previous classes had taken weeks or months to build. After the class, as the instructors sat processing what just happened, we realized there's no going back.

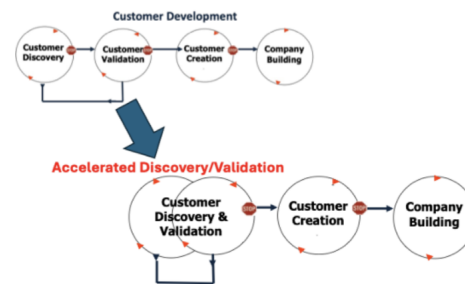
I've been writing about how AI is going to change startups (<https://steveblank.com/2026/03/17/your-startup-is-probably-dead-on-arrival/>), but the shock of seeing eight teams actually implementing it was mind-blowing. And not a single team thought they were doing anything extraordinary.

## Product Development Velocity is Off the Scale

The old sequence for our class was simple: We had teams replicate what they would do in a startup. Have an idea. Build a team. Get out of the building to talk to customers to understand their problems. Do Agile development and DevSecOps to build MVPs over 10 weeks to test the solutions. And if they were going to build a company, discover and develop a "moat" of proprietary code and features.

This year, in the first week of the class, our students used multiple AI tools to replace what previously would

have taken a large development team. They used [Perplexity](https://www.perplexity.ai/) for research, [Claude Code](https://code.claude.com/docs/en/overview) (<https://code.claude.com/docs/en/overview>) and [Replit](https://replit.com/) (<https://replit.com/>) to build apps, [Vercel](https://vercel.com/) (<https://vercel.com/>) / [v0](https://v0.app/) (<https://v0.app/>) for prototyping, (<https://v0.app/>) [Granola](https://www.granola.ai/) (<https://www.granola.ai/>) to auto-transcribe and summarize customer interviews. The whole flow was compressed.



Because it was so easy to have an idea and then build something in minutes/hours, our students showed up on the first day of the class with products. They no longer had to wait weeks or months before testing whether anyone cares.

We realized we were watching a massive acceleration of the Customer Discovery/Customer Validation timeline.

## Class Observations

### Learning 1: Impedance Mismatch Between Product Development and Learning

By the third week of the class we observed that the velocity of product development meant that teams could now generate more products than they could validate. The amount of product did not equal the amount of learning. Teams were so overwhelmed with so much information from the AI tools that they lost sight of the goal of customer development. They started to believe

that the product itself was the truth.

The upshot is that AI has made customer validation harder. The abundance and ease of creating MVPs has become an accidental **denial of service attack** (<https://www.cloudflare.com/learning/ddos/glossary/denial-of-service/>) on the search for a repeatable and scalable business model. While this is an artifact of today, it means we need a different model for customer development as rapid coding isn't going away.

### Learning 2: Student dependence on ChatGPT decreased the quality of insights.

After week two of the class, it was clear teams were delegating communication to an AI. This dumbed down communication turned into **AI slop** ([https://en.wikipedia.org/wiki/AI\\_slop](https://en.wikipedia.org/wiki/AI_slop)). ChatGPT and Claude are no substitute for thoughtful communication - whether it's email, PowerPoint or weekly summaries of Lessons Learned. Luckily you can spot this quickly.

### Learning 3: Customers are feeling disrupted.

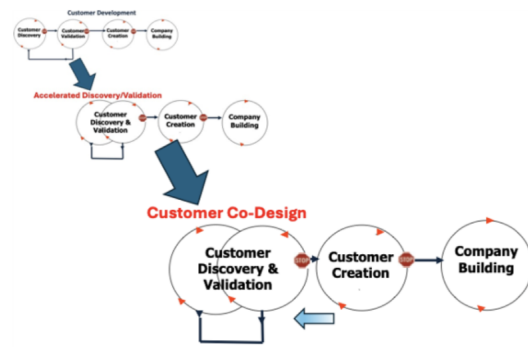
As the student teams got out of the building, they discovered that potential customers were already feeling disrupted by AI. Many of the companies the teams demo'd to realized that they were seeing not just incremental improvements, but in fact were being shown a "going out of business" scenario.

### Learning 4: Customers realize their proprietary data might be their only moat.

In some cases, potential customers who would have previously shared their data with students are now asking for NDAs to share information with the team. Customers are realizing that closely held and hard-won information might be one of the few barriers to AI.

### Potential 1: Customer Co-Design

As AI tools are allowing our teams to build higher fidelity MVPs, a few are beginning to consider using the MVPs as digital twins (as a simulation of the final product). When put in the cloud and shared with potential earlyvangelists, startups can now start co-designing the product with potential prospects. Teams can monitor if the digital twin is being used, how it's used, and the feedback of what features are needed can be shared instantly. Teams can update the digital twin as they add features.



### Potential 2: Agent/Customer Outcome Fit

Today, software applications are built to give users information and then expect the users to do the work via a user interface of dashboards, alerts, workflow tools, and reports. But customers buy software to get a job done, not to look at more screens. Getting the job done is what AI Agents (orchestrated by tools like **OpenClaw** (<https://openclaw.ai/>)) will autonomously enable. For some teams, future class sections may see the search for Product/Market fit become the search for AI Agent/Customer Outcome fit. Minimum Viable Products (MVPs) will become Minimum Productive Outcomes (MPOs).

## Lessons Learned

- **MVPs are no longer an indication of technical competence.** Vibe coding has transformed MVPs to the equivalent of PowerPoint slides
- **Speed to MVPs hasn't yet meant faster learning about building a company.** While we're still early in the class, the blinding speed of the first week's onslaught of MVPs hasn't yet translated into faster learning about customer validation.
- **Business process and business models still matter.** The bottleneck for our student teams has moved from needing a fraction of the resources to build high-quality MVPs to judgment: how to choose the right problem, how to read user signals correctly, and deciding what to build next.
- **Product/market fit and agent/outcome fit will co-exist (for a while).** While some customers are ready to move to an Agentic workflow, for others delivering Product/Market Fit is still what users want to see.

- **Startup teams will be smaller.** Our class teams are 4-5. In the past, if they decided to pursue their idea and start a company they would need to hire a larger team to build the product, manage the product, find out whether they had product/market fit, create demand, etc. That's mostly no longer true. Most teams won't need to raise money to find out if the problem is real or before they know if users care.
- **Enterprise pricing models will change.** Some teams are already testing pricing that will shift from per/seat to workflows, outcomes, results, resolutions, successful tasks.
- **Customer development will change.** Because the Customer Development cycle is faster and multiple MVPs now can be run simultaneously
- Effort shifts to the extra time needed on hypotheses testing because the velocity and volume of product development can overwhelm signals from potential customers.
- **As MVPs rapidly change,** they need to be instrumented to monitor customer usage/interactions.