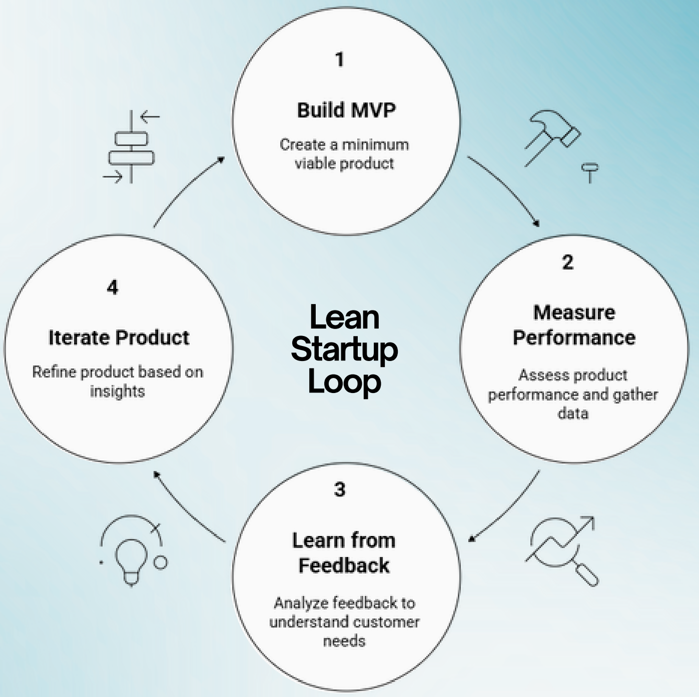
What is a Minimum Viable Product (MVP)?

MVP

A Minimum Viable Product (MVP) is the most basic version of a product that includes just enough features to satisfy early adopters and gather validated learning about users and the market.



1 Build MVP

Create a minimum viable product

2 Measure Performance

Assess product performance and gather data

3 Learn from Feedback

Analyze feedback to understand customer needs

4 Iterate Product

Refine product based on insights

Lean Startup Loop

An MVP is not the smallest product you can build; it's the fastest way to start the learning process.

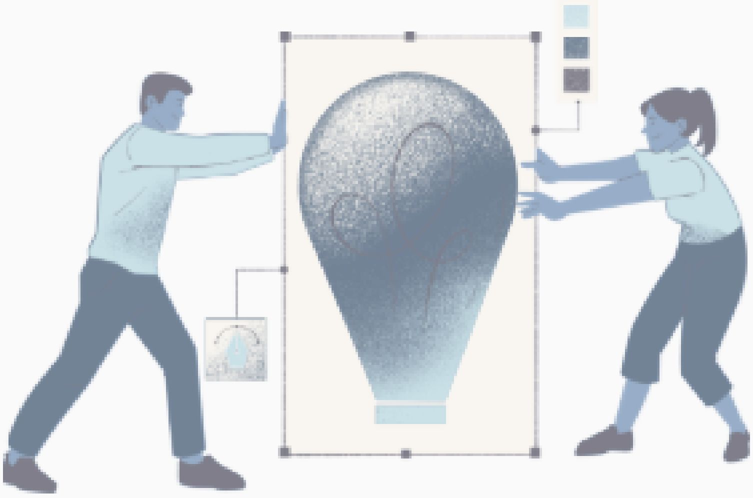
Importance of MVP:

- Validates product-market fit before large investments.

- Reduces time-to-market and avoids overbuilding.

- Allows early feedback from real users.

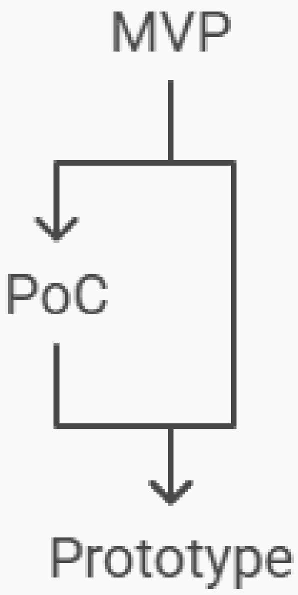
- Helps prioritize core features based on actual user behavior.



Start simple, learn fast, grow smart.

MVP vs PoC vs Prototype

Feature	PoC	Prototype	MVP
Purpose	Validate feasibility	Explore look/feel	Validate real user need
Audience	Internal/ Investors	Stakeholders/ Designers	Early adopters/user
Functionality	Very limited	Simulated/basic functionality	Basic working product
Outcome	Technical proof	Design exploration	Market learning



Key Characteristics of an MVP

- Solves one core problem well

- Quick to build and launch
- Testable with real users
- Iterative in nature
- Includes only must-have features, not nice-to-haves

“If you're not embarrassed by your first product release, you’ve launched too late.” – Reid Hoffman

Steps to Build an MVP

7 actionable steps from user identification to iteration

1 Identify the target users and core problem

2 Define the value proposition clearly.

3 List all possible features, then prioritize the essentials.

4 Design and develop the simplest version

5 Launch to a small user group or early adopters

6 Collect feedback and usage data

7 Iterate based on learnings

Real-World MVP Examples:

Airbnb:

Rented out a room in their own apartment to validate demand.

Dropbox:

Started with a simple explainer video before building the full product.

Uber:

Initial version worked only for iPhones and operated only in San Francisco.

Common Mistakes to Avoid

- Trying to build a “perfect” product on first attempt
- Overloading MVP with too many features
- Ignoring feedback from early users
- Skipping testing in real conditions
- MVP is about learning, not launching

Your MVP should be simple enough to launch quickly, but useful enough to be valuable.