

Lean Product & Process Development Starter Kit



What is Lean Product & Process Development (LPPD)?

LPPD is a systems approach for creating new products, services, and the processes that deliver them, with the aim of maximizing customer value while minimizing waste across the whole development value stream.

It treats development as a high-stakes learning job: front-loading knowledge, delaying irreversible decisions, and designing products and production systems together.

Guiding Principles

The Six Pillars of Lean Product & Process Development

These principles form the foundation of how we develop products and the systems that deliver them. They come from lean thinking but are adapted for the unique challenges of development work—where uncertainty is high, knowledge is the currency, and people drive innovation.

Principle 1: Put People First

Develop the developers so they can develop great products.

Lean development recognizes that people—not processes or tools—create value. Design your system so engineers, designers, and cross-functional teams can do their best work: give them the knowledge, authority, and environment to solve problems and make decisions. When you respect people’s expertise and empower them to improve both product and process, you unlock innovation and ownership.

Principle 3: Develop as a Team Sport

Cross-functional collaboration beats sequential hand-offs.

Development isn’t a relay race where design throws to engineering who throws to manufacturing. Lean development is simultaneous, collaborative work with shared ownership of outcomes. Bring customer insight, design, engineering, manufacturing, and service perspectives together from concept to launch—and keep them synchronized through visual management and rhythmic communication.

Principle 5: Build in Learning and Knowledge Reuse

Make learning visible, capture it, and use it again. Every development project generates valuable knowledge—about customers, technologies, manufacturing, failures, and trade-offs—but in traditional systems, this knowledge stays tacit and is lost. Lean development makes learning explicit through A3s, trade-off curves, design standards, and lessons-learned reviews. Reusing knowledge is a core competitive advantage: it accelerates future projects and reduces repeated mistakes.

Principle 2: Understand Before Executing

Front-load knowledge, back-load decisions.

In traditional development, we commit to solutions early and discover problems late. Lean development flips this: invest time up front to deeply understand customers, context, constraints, and trade-offs. Use rapid experiments, set-based exploration, and visual modeling to build knowledge cheaply before locking in costly decisions.

Principle 4: Synchronize Workflows

Establish cadence and “good enough” interfaces so work can flow.

Instead of waiting for perfect information, define stable points where teams can exchange “good enough” knowledge and work concurrently. Use cadenced cycles (sprints, integration events, design reviews) to create predictable rhythm and reduce waiting. This enables flow even in high-uncertainty environments and prevents the chaos of everyone working in isolation.

Principle 6: Design the Value Stream

Optimize the whole, not the parts.

Lean development isn’t just about designing a great product; it’s about designing the product AND the process to deliver it as an integrated system. Make decisions that improve flow, quality, and speed across the entire value stream—from customer insight through manufacturing ramp-up and service—not just what’s best for one department. This principle prevents local optimization that creates bottlenecks or waste downstream.



Development value stream: end-to-end flow of work and knowledge, powered by lean learning cycles



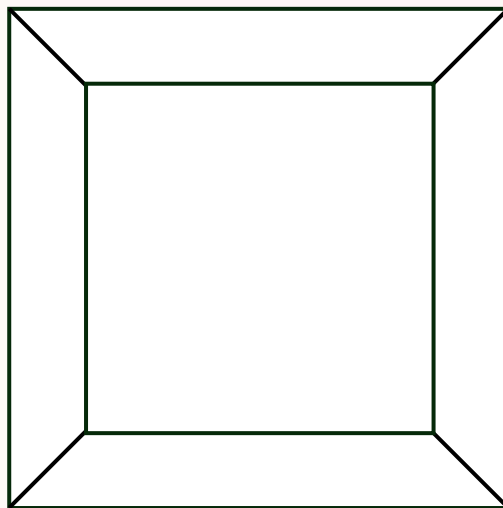
- 1** STRATEGY & BETS
True North
Value Streams
Portfolio Bets
- 2** DISCOVER
Customer Problems
System Context
Gemba
- 3** DESIGN & FLOW
Architecture
Dev Value Stream Design
Obeya
- 4** LEAN LEARNING CYCLES
Set-Based Design Experiments
Prototyping Ladder
Knowledge-Based Milestones
- 5** INDUSTRIALIZATION
Design 3P
Process 3P
Production 3P
Ready-to-Run Systems
- 6** SCALING & LIFECYCLE
Ramp-Up
Variants
Roadmap
Learning to Next Bets

The minimal Obeya layout

North / Strategy Wall

True North and key development value streams.

Target conditions and active portfolio bets.



Learning & Issues Wall

Current questions, experiments, and K-Briefs in focus.

Major issues, owners, and agreed next steps.

Development Value Stream Flow Wall

Map of the development value stream from idea to launch.

Active bets plotted on the stream, WIP limits, simple status signals (e.g., “on track / at risk / blocked”).

Design Options Wall

Set-based options for product and process (concepts, architectures, process routes).

Key experiments and decisions needed to narrow the sets.

How to Use This Starter Kit (Next 4–6 Weeks)

1. Pick one value stream or product family.
 - Choose a real development flow (e.g., a platform or product family) rather than “everything”.
2. Sketch your development value stream.
 - Use the LPPD spine (Strategy & Bets → Discover → Design & Flow → Lean Learning Cycles → Industrialization (3P) → Scaling & Lifecycle).
 - Draw it as a simple line with boxes and place it on the Development Value Stream Flow Wall.
3. Build a lightweight Obeya using the four walls.
 - For each wall, add just enough content to start:
 - North / Strategy: 1–2 True North statements, 2–3 target conditions, 3–5 main bets.
 - Flow: the value stream sketch, current bets, simple WIP limit (e.g., max 3 active bets).
 - Design Options: a few key concept/process alternatives and the experiments you’re planning.
 - Learning & Issues: current cycles, K-Briefs in focus, top 3 issues and owners.
 - Example: pick your current platform, map its development flow, and use the Obeya as the weekly focal point for that platform’s decisions.
4. Choose one principle to focus on first.
 - For example:
 - “Understand Before Executing” → run one Lean Learning Cycle and make its experiments and K-Brief visible on the Learning & Issues wall.
 - “Design the Value Stream” → refine the development value stream map and use the Flow wall in weekly check-ins.

Use this Obeya as a weekly conversation space: walk the walls, update them lightly, and ask ‘What did we learn?’ and ‘What’s blocking flow?’ before you talk about dates and tasks.

For a full treatment of these concepts, see the Lean PPD Playbook at leanpeakproductlab.com/playbook