



# Módulo 2. Diseño de experimentos

☰ Diseño de experimentos

☰ Iteración y aprendizaje

☰ Referencias

# Diseño de experimentos

---

En el desarrollo de productos, la construcción del **producto mínimo viable (MVP)** no constituye únicamente una instancia de lanzamiento, sino que se comprende como un **instrumento de aprendizaje y validación** dentro de contextos de alta incertidumbre. En este sentido, distintos enfoques coinciden en que el MVP debe interpretarse más como un **experimento que como un producto terminado**, ya que su principal objetivo es **poner a prueba hipótesis sobre el problema, la solución y el mercado** (Bassino, 2016).

Esta perspectiva se inscribe dentro de la metodología *Lean Startup*, la cual propone que todo emprendimiento es, en esencia, **un conjunto de supuestos que deben ser validados empíricamente** antes de escalar una solución. En lugar de desarrollar productos completos desde el inicio, este enfoque prioriza la creación de versiones simplificadas que permitan **aprender lo más rápido y con el menor costo posible** (Lean Case, s.f.). De este modo, el desarrollo de productos deja de ser

un proceso lineal para convertirse en un **proceso iterativo basado en evidencia**.

En esta línea, el MVP se define como **la versión más simple de un producto capaz de generar aprendizaje validado a partir del contacto con usuarios reales**, permitiendo obtener retroalimentación sobre sus necesidades, comportamientos y preferencias (Caneva, 2022) . A su vez, su valor no reside únicamente en la funcionalidad que ofrece, sino en su capacidad para **reducir la incertidumbre y minimizar riesgos antes de invertir mayores recursos** (Infinitia, 2024) .

Sin embargo, para que este aprendizaje sea significativo, no alcanza con lanzar un MVP y observar resultados de manera informal. Es necesario avanzar hacia una lógica más estructurada donde las decisiones se sustenten en evidencia. Aquí es donde cobra relevancia el diseño de experimentos, entendido como el proceso mediante el cual los equipos definen **qué quieren aprender, cómo lo van a medir y bajo qué condiciones van a validar sus hipótesis**.

En este contexto, el desarrollo de productos se articula con el ciclo *build-measure-learn*, donde cada iteración implica **construir una solución, medir su desempeño y aprender a partir de los datos obtenidos**. Este ciclo permite transformar la intuición en conocimiento validado, facilitando la toma de decisiones

informadas y reduciendo el margen de error en etapas tempranas del desarrollo.

Por lo tanto, el diseño de experimentos no solo complementa al MVP, sino que lo potencia, ya que convierte la interacción con los usuarios en una **fuentes sistemática de aprendizaje**. A lo largo de esta unidad, se abordarán los principios fundamentales para formular hipótesis, identificar supuestos críticos y seleccionar métodos de testeo adecuados, con el objetivo de **validar propuestas de valor de manera rigurosa y orientada a resultados**.

## Formulación de hipótesis y definición de supuestos críticos

Una vez comprendido que el **MVP funciona como un experimento orientado al aprendizaje**, el siguiente paso en el diseño de experimentos consiste en **explicitar aquello que se desea validar**. En este contexto, los equipos no trabajan sobre certezas, sino sobre **supuestos** acerca del problema, la solución y el comportamiento de los usuarios. Por ello, resulta fundamental transformar esas ideas iniciales en **hipótesis claras, comprobables y medibles**.

En el marco de la metodología *Lean Startup*, los emprendimientos operan bajo condiciones de **incertidumbre extrema**, lo que implica que muchas de las decisiones iniciales se basan en intuiciones o creencias no verificadas. La formulación de hipótesis permite **ordenar esa incertidumbre**, convirtiendo percepciones generales en afirmaciones específicas que pueden ser puestas a prueba mediante experimentos (Lean Case, s.f.).

En términos operativos, una hipótesis establece una relación entre variables y anticipa un resultado esperado. Por ejemplo, un equipo puede plantear que, **si se ofrece una solución determinada, entonces los usuarios realizarán una acción específica**, como registrarse, comprar o interactuar con una funcionalidad. Este tipo de formulaciones permite diseñar experimentos orientados a validar o refutar dichas suposiciones.

En este sentido, el MVP cumple un rol central, ya que actúa como el medio a través del cual estas hipótesis se contrastan con la realidad. Tal como se señala en el material, su propósito no es validar un producto final, sino **generar aprendizaje validado sobre las necesidades y comportamientos de los usuarios** (Infinitia, 2024) . Esto implica que cada funcionalidad o decisión de diseño debería responder a una hipótesis previa.

Ahora bien, no todos los supuestos tienen el mismo nivel de relevancia. Por ello, es necesario identificar los **supuestos**

**críticos**, es decir, aquellos que, en caso de ser incorrectos, comprometerían la viabilidad del proyecto. Estos suelen vincularse con tres dimensiones principales:

- **Supuestos sobre el problema:** ¿Existe realmente una necesidad o dolor relevante para los usuarios?
- **Supuestos sobre la solución:** ¿La propuesta diseñada resuelve efectivamente ese problema?
- **Supuestos sobre el mercado:** ¿Los usuarios estarían dispuestos a adoptar o pagar por la solución?

La identificación de estos supuestos permite priorizar los experimentos, enfocando los esfuerzos en **reducir los mayores niveles de incertidumbre** en las etapas iniciales. De este modo, en lugar de invertir recursos en desarrollar funcionalidades completas, los equipos pueden validar progresivamente los aspectos más críticos del modelo.

Asimismo, la formulación de hipótesis debe estar acompañada por la definición de **criterios de éxito y métricas**, que permitan interpretar los resultados de los experimentos. En este punto, cobra relevancia la capacidad de medir comportamientos concretos de los usuarios, ya que el aprendizaje en el enfoque *lean* se basa en evidencia observable y no en opiniones o percepciones subjetivas.

En síntesis, la formulación de hipótesis y la identificación de supuestos críticos constituyen una etapa clave en el diseño de experimentos, ya que permiten transformar la incertidumbre en un proceso estructurado de validación. A partir de este enfoque, el desarrollo de productos se orienta hacia la toma de decisiones informadas, donde cada iteración se fundamenta en el aprendizaje obtenido a través de la experimentación.

## Diseño de experimentos: tipos de test (A/B, smoke test, entrevistas, etc.)

Una vez definidas las **hipótesis y los supuestos críticos**, el siguiente paso consiste en diseñar experimentos que permitan validarlos de manera concreta. En este punto, el foco ya no está solo en qué se quiere aprender, sino en **cómo generar evidencia confiable para tomar decisiones**.

Dentro del enfoque *Lean Startup*, el diseño de experimentos implica seleccionar **métodos de testeo adecuados al nivel de incertidumbre y al tipo de hipótesis** que se desea validar. No

todos los experimentos requieren el mismo nivel de desarrollo: en muchos casos, es posible aprender con soluciones simples, como prototipos, simulaciones o incluso representaciones parciales de la propuesta (Bassino, 2016).

En este sentido, el MVP no es una única forma de producto, sino que puede adoptar distintos formatos según el objetivo del experimento. Tal como se observa en el material, puede ir desde **landing pages hasta prototipos o pruebas simuladas**, siempre con el objetivo de **obtener aprendizaje validado con el menor esfuerzo posible** (Washington, 2024).

A continuación, se presentan algunos de los principales tipos de test utilizados en el diseño de experimentos:

1

### **Test A/B (A/B testing)**

Consiste en comparar dos versiones de una misma solución para evaluar cuál genera mejores resultados en función de una métrica específica. Por ejemplo, se pueden testear dos versiones de una interfaz o mensaje para observar cuál logra mayor conversión. Este tipo de experimento permite **aislar variables y medir su impacto directo en el comportamiento del usuario**, siendo especialmente útil en entornos digitales.

---

2

### ***Smoke test***

El *smoke test* se utiliza para validar el interés del mercado antes de construir el producto. Generalmente, se presenta una propuesta (por ejemplo, una *landing page*) como si el producto existiera, midiendo cuántos usuarios muestran intención de uso o compra. Este enfoque permite **evaluar la demanda sin necesidad de desarrollar la solución completa**, reduciendo significativamente el riesgo (Infinitia, 2024).

3

### **Entrevistas a usuarios**

Las entrevistas permiten obtener información cualitativa sobre las necesidades, motivaciones y problemas de los usuarios. A diferencia de otros métodos, no buscan validar comportamientos medibles, sino **comprender en profundidad el contexto del usuario**, lo cual resulta clave en etapas tempranas. Este tipo de test es especialmente útil para validar **supuestos sobre el problema**.

4

### **Prototipos y pruebas de usabilidad**

Los prototipos permiten representar una versión simplificada del producto para observar cómo los usuarios interactúan con él. Pueden variar en fidelidad, desde bocetos hasta versiones más avanzadas. Su principal valor radica en que permiten **detectar problemas de uso, fricciones**

**y oportunidades de mejora** antes de invertir en desarrollo completo (Caneva, 2022).

5

### **Experimentos tipo *Wizard of Oz***

En este tipo de test, el usuario cree interactuar con un sistema automatizado, pero en realidad hay una intervención manual detrás. Esto permite simular funcionalidades complejas sin necesidad de desarrollarlas completamente, facilitando la validación de la propuesta con un costo reducido.

La selección del tipo de experimento depende de múltiples factores, como el nivel de avance del producto, el tipo de hipótesis y los recursos disponibles. Sin embargo, todos estos métodos comparten un mismo principio: **aprender lo máximo posible con la mínima inversión.**

Antes de seleccionar un tipo de experimento, es fundamental comprender que **cada método de testeo presenta fortalezas y limitaciones específicas**, en función del tipo de hipótesis que se desea validar y del nivel de desarrollo del producto. En este sentido, no existe un test “mejor” en términos absolutos, sino que su pertinencia depende del **objetivo de aprendizaje, los recursos disponibles y el grado de incertidumbre.**

A continuación, se presenta una comparación de los principales tipos de test utilizados en el diseño de experimentos, destacando sus aspectos positivos y negativos para facilitar su aplicación estratégica:

**Tabla 1: Ventajas y limitaciones de los principales tipos de test en la validación de MVP**

Tipo de test	Aspectos positivos	Aspectos negativos
<b>Test A/B (A/B testing)</b>	Permite <b>medir con precisión el impacto de cambios específicos.</b> Genera <b>datos cuantitativos confiables.</b> Facilita la <b>toma de decisiones basada en métricas.</b>	Requiere <b>volumen de usuarios significativo.</b> No explica el “por qué” detrás del comportamiento. Puede ser complejo de implementar en etapas tempranas.
<b>Smoke test</b>	Permite <b>validar interés del mercado sin</b>	Puede generar <b>falsos positivos</b> (interés sin

	<p><b>desarrollar el producto.</b></p> <p>Reduce costos y riesgos.</p> <p>Es rápido de ejecutar.</p>	<p>intención real de uso).</p> <p>No valida la experiencia real del producto.</p>
<p><b>Entrevistas a usuarios</b></p>	<p>Brindan <b>insights profundos y cualitativos.</b></p> <p>Permiten comprender <b>motivaciones, problemas y contextos.</b></p> <p>Ideales para etapas iniciales.</p>	<p>Resultados <b>subjetivos y difíciles de generalizar.</b></p> <p>Dependen de la interpretación del investigador.</p> <p>No miden comportamiento real.</p>
<p><b>Prototipos y test de usabilidad</b></p>	<p>Permiten <b>detectar problemas de interacción y diseño.</b></p> <p>Facilitan iteraciones rápidas.</p> <p>Acercan la solución</p>	<p>Requieren <b>tiempo de diseño y preparación.</b></p> <p>Pueden no reflejar completamente la experiencia final.</p>

	a un contexto real de uso.	
<b><i>Wizard of Oz</i></b>	<p>Permite <b>simular funcionalidades complejas sin desarrollarlas.</b></p> <p>Reduce costos técnicos iniciales.</p> <p>Valida la propuesta de valor rápidamente.</p>	<p>Puede ser <b>difícil de escalar.</b></p> <p>Riesgo de sesgos si la simulación no es consistente.</p> <p>Requiere intervención manual constante.</p>

**Fuente:** Elaboración propia en base a Bassino (2016); Caneva (2022); Infinitia (2024); Washington (2024) .

Esta comparación permite entender que el diseño de experimentos implica **combinar distintos tipos de test a lo largo del proceso**, pasando de métodos exploratorios (como entrevistas) a otros más estructurados y cuantitativos (como *A/B testing*), en función del aprendizaje que se busca generar en cada etapa.

En este marco, el diseño de experimentos no debe entenderse como una instancia aislada, sino como parte de un proceso continuo donde cada test contribuye a reducir la incertidumbre.

A través de estos experimentos, los equipos pueden avanzar desde supuestos iniciales hacia **conocimiento validado**, construyendo productos de manera más eficiente y alineada con las necesidades reales de los usuarios.

## Priorización de hipótesis en contextos de múltiples supuestos críticos

En el desarrollo de productos, especialmente bajo el enfoque *Lean Startup*, es habitual que un equipo enfrente **múltiples supuestos críticos de manera simultánea**. Estos supuestos pueden vincularse con el problema, la solución o el mercado, y todos presentan cierto nivel de incertidumbre. En este escenario, la priorización de hipótesis se convierte en una decisión estratégica que permite organizar el proceso de validación de forma eficiente.

El primer criterio para priorizar hipótesis es el **nivel de riesgo asociado a cada supuesto**. Se consideran prioritarias aquellas hipótesis que, en caso de ser incorrectas, comprometerían la viabilidad del producto. Por ejemplo, validar si el problema realmente existe o si las personas están dispuestas a utilizar la solución resulta más urgente que optimizar aspectos secundarios del producto. Esta lógica permite enfocar los esfuerzos en **reducir la incertidumbre más crítica en etapas tempranas**.

Un segundo criterio es el **impacto sobre la propuesta de valor**. Las hipótesis directamente vinculadas con el beneficio central del producto deben testearse antes que aquellas relacionadas con mejoras o extensiones. Esto implica identificar qué supuestos sostienen la relación entre problema y solución, y priorizar su validación para asegurar que el producto tenga sentido en el contexto de uso.

Asimismo, resulta relevante considerar el **nivel de evidencia disponible**. Las hipótesis que cuentan con menor respaldo empírico requieren ser validadas con mayor urgencia. En cambio, aquellas que ya poseen algún grado de validación previa pueden postergarse. Este criterio permite avanzar desde lo más incierto hacia lo más consolidado, organizando el aprendizaje de manera progresiva.

Otro aspecto a tener en cuenta es el **costo y la rapidez de validación**. En el enfoque experimental, se priorizan aquellas hipótesis que pueden testearse con menor esfuerzo y en menor tiempo, siempre que aporten aprendizaje significativo. Esta lógica se alinea con el principio de **aprender lo máximo posible con la mínima inversión**, evitando desarrollos innecesarios.

**Tabla 2. Criterios para priorizar hipótesis en el diseño de experimentos**

<b>Criterio</b>	<b>Descripción</b>	<b>Aplicación</b>
<b>Riesgo</b>	Impacto del supuesto en la viabilidad del producto	Priorizar lo que puede invalidar el proyecto
<b>Impacto en valor</b>	Relación con la propuesta de valor	Testear el núcleo de la solución
<b>Nivel de evidencia</b>	Grado de validación previa	Validar primero lo más incierto
<b>Costo y rapidez</b>	Esfuerzo necesario para testear	Elegir experimentos eficientes

Fuente: elaboración propia con base en enfoque *Lean Startup* (Lean Case, s.f.; Infinitia, 2024).

Desde una perspectiva operativa, estos criterios no se aplican de forma aislada, sino de manera combinada. Los equipos suelen organizar sus hipótesis en matrices o listas priorizadas, donde se cruzan variables como riesgo e impacto para definir un orden de validación. Este enfoque permite estructurar el proceso experimental y evitar decisiones basadas únicamente en intuición.

En síntesis, la priorización de hipótesis permite transformar un conjunto amplio de supuestos en un **plan de validación ordenado**, donde cada experimento responde a una necesidad concreta de aprendizaje. A través de este proceso, el desarrollo de productos se orienta hacia la **toma de decisiones basada en evidencia**, reduciendo la incertidumbre de manera progresiva y facilitando la construcción de soluciones ajustadas a contextos reales.

CONTINUAR

# Iteración y aprendizaje

---

En el desarrollo de productos, el diseño de experimentos permite estructurar la validación de hipótesis y obtener información a partir del comportamiento de los usuarios. Sin embargo, el valor de este proceso no se agota en la ejecución de pruebas aisladas, sino que se consolida en la capacidad de los equipos para **interpretar los resultados y tomar decisiones a partir de la evidencia obtenida**. En este punto, el proceso se orienta hacia una lógica de iteración, donde cada resultado se integra como insumo para redefinir el rumbo del producto.

En contextos profesionales, esta etapa se presenta cuando los equipos deben decidir cómo continuar luego de haber testeado una hipótesis: mantener la dirección actual, realizar ajustes o reformular la propuesta. Estas decisiones no se basan en intuiciones, sino en el análisis sistemático de los datos recolectados durante los experimentos. De este modo, el desarrollo del producto se organiza como un proceso dinámico, donde cada iteración incorpora aprendizajes que permiten reducir la incertidumbre.

La iteración se vincula directamente con el ciclo *build-measure-learn*, que propone una secuencia continua de construcción, medición y aprendizaje. Este ciclo permite que el producto evolucione a partir de evidencia empírica, integrando tanto datos cuantitativos como cualitativos en la toma de decisiones. En lugar de avanzar hacia una versión final de manera lineal, el producto se ajusta progresivamente en función de los resultados obtenidos en cada instancia de validación.

Asimismo, esta etapa introduce una dimensión estratégica en el proceso de desarrollo. No todos los resultados implican la misma decisión: algunos validan hipótesis y permiten avanzar, otros indican la necesidad de realizar ajustes, y otros sugieren la conveniencia de redefinir aspectos centrales del producto. La capacidad de interpretar estos escenarios resulta necesaria para sostener un proceso de aprendizaje continuo y orientado a resultados.

A lo largo de esta unidad, abordaremos cómo se recolecta y analiza el *feedback* obtenido en los experimentos, y cómo este aprendizaje se integra en el ciclo iterativo de desarrollo, permitiendo tomar decisiones fundamentadas en evidencia en contextos reales de uso.

# Recolección de feedback cualitativo y cuantitativo

En el proceso de validación de un **producto mínimo viable (MVP)**, la recolección de datos constituye una instancia central para transformar la experimentación en aprendizaje. En este contexto, el feedback obtenido a partir de la interacción con los usuarios permite **evaluar hipótesis, identificar problemas y orientar decisiones de mejora**. Para ello, se utilizan dos enfoques complementarios: el **feedback cualitativo** y el **feedback cuantitativo**, cada uno con características, alcances y aplicaciones específicas dentro del desarrollo de productos.

El **feedback cualitativo** se orienta a comprender la experiencia del usuario desde una perspectiva interpretativa. Se obtiene a través de técnicas como entrevistas en profundidad, observaciones, pruebas guiadas o conversaciones abiertas con usuarios. Su principal característica es que permite acceder a **significados, percepciones, emociones y motivaciones**, elementos que no pueden captarse mediante datos numéricos.

Desde el punto de vista operativo, este tipo de *feedback* resulta especialmente útil en etapas iniciales del desarrollo, cuando el equipo necesita comprender si el problema identificado es relevante y cómo se manifiesta en la vida cotidiana de las personas. Por ejemplo, a través de entrevistas, un equipo puede

detectar que los usuarios no utilizan una aplicación no por falta de interés, sino porque no comprenden su funcionamiento o no perciben su utilidad inmediata. Este tipo de información permite **reformular la propuesta de valor o ajustar la experiencia de uso**.

Entre sus principales características, el *feedback* cualitativo se distingue por ser **profundo, contextual y exploratorio**. Permite analizar situaciones complejas, identificar patrones de comportamiento no evidentes y descubrir necesidades que no habían sido consideradas en el diseño inicial. Sin embargo, su interpretación requiere un análisis cuidadoso, ya que los resultados no se expresan en términos generalizables, sino como insights que deben ser contrastados con otros datos.

**Tabla 7. Características del *feedback* cualitativo**

<b>Aspecto</b>	<b>Descripción</b>	<b>Ejemplo de aplicación</b>
Tipo de datos	Narrativos, descriptivos	Opiniones, relatos, experiencias
Método	Entrevistas, observación, testeo	Entrevistas a usuarios sobre uso

	guiado	del producto
Objetivo	Comprender el comportamiento y contexto	Detectar problemas de uso o percepción
Resultado	Insights y patrones interpretativos	Identificación de necesidades no explícitas

**Fuente:** Elaboración propia con base en Infinitia Industrial Consulting (2024).

Por otro lado, el **feedback cuantitativo** se basa en la recolección de datos medibles que permiten analizar el comportamiento del usuario en términos numéricos. Este tipo de información se obtiene a través de métricas como tasas de conversión, cantidad de usuarios activos, tiempo de uso, retención o abandono en determinadas etapas del proceso.

**Su principal característica es que permite identificar patrones de comportamiento a gran escala, facilitando la comparación entre distintas versiones del producto o entre distintos grupos de usuarios. Por ejemplo, un equipo puede medir cuántas personas completan un registro en una**

plataforma y detectar en qué punto del proceso se produce el abandono. Esta información permite identificar problemas específicos en el flujo de uso y priorizar mejoras.

En términos operativos, el *feedback* cuantitativo resulta especialmente útil en etapas donde el producto ya se encuentra en uso y es posible recolectar datos de interacción. A diferencia del enfoque cualitativo, este tipo de *feedback* permite **validar hipótesis con mayor nivel de precisión**, ya que se apoya en indicadores concretos.

**Tabla 8. Características del *feedback* cuantitativo**

Aspecto	Descripción	Ejemplo de aplicación
Tipo de datos	Numéricos, medibles	Porcentajes, tasas, métricas
Método	Analítica, métricas de uso, test A/B	Medición de conversión en una <i>landing page</i>

Objetivo	Identificar patrones y tendencias	Detectar abandono en un proceso
Resultado	Datos estadísticos	Comparación entre versiones del producto

**Fuente:** Elaboración propia con base en Washington (2024).

Desde una perspectiva metodológica, ambos tipos de *feedback* no se excluyen, sino que se complementan. Mientras el enfoque cuantitativo permite responder **qué está ocurriendo en el producto**, el enfoque cualitativo permite comprender **por qué ocurre**. Esta articulación resulta necesaria para construir una interpretación completa del comportamiento del usuario.

Por ejemplo, si un equipo detecta que una gran cantidad de usuarios abandona un proceso de registro (dato cuantitativo), puede recurrir a entrevistas o pruebas de usabilidad para comprender qué dificultades encuentran en ese punto del recorrido (dato cualitativo). De este modo, la combinación de ambos enfoques permite pasar de la observación del problema a la comprensión de sus causas.

En el ejercicio profesional, la recolección de *feedback* requiere definir previamente **qué información se necesita, cómo se va a obtener y cómo se va a analizar**. Esta planificación permite

evitar la acumulación de datos sin propósito y asegura que cada instancia de recolección contribuya al proceso de validación.

En síntesis, la recolección de *feedback* cualitativo y cuantitativo constituye una dimensión central en el desarrollo de productos, ya que permite transformar la interacción con el MVP en conocimiento. A través de sus características, aplicaciones y complementariedad, estos enfoques permiten construir una base sólida para la toma de decisiones basada en evidencia, orientando el proceso hacia soluciones ajustadas a contextos reales de uso.

## Ciclo build-measure-learn y toma de decisiones basada en evidencia

En el enfoque *Lean Startup*, el desarrollo de productos se organiza a partir del ciclo *build-measure-learn*, el cual estructura el proceso de validación en una secuencia iterativa orientada al aprendizaje. Este ciclo establece una dinámica donde cada acción de desarrollo se vincula con la generación de evidencia, permitiendo que las decisiones se fundamenten en datos y no en suposiciones (Ries, 2011; EAE Business School, s.f.).

El ciclo comienza con la fase ***build***, donde se construye una versión del producto —generalmente un **MVP**— diseñada para

poner a prueba una hipótesis específica. En esta etapa, el objetivo no consiste en desarrollar una solución completa, sino en crear una versión funcional que permita generar interacción con usuarios reales. Esta construcción se realiza a partir de las decisiones previamente definidas en el diseño de experimentos.

La segunda fase corresponde a *measure*, donde se recolectan datos a partir de la interacción con el producto. En esta instancia, se aplican los métodos de recolección de *feedback* cualitativo y cuantitativo, permitiendo observar comportamientos, registrar métricas y analizar la experiencia del usuario. Tal como se plantea en los materiales, esta medición resulta necesaria para transformar la interacción en información útil para la validación (Washington, 2024).

La tercera fase es *learn*, donde los datos recolectados se interpretan en función de las hipótesis planteadas. En este punto, el equipo analiza si los resultados obtenidos validan o refutan los supuestos iniciales. Este análisis permite generar **aprendizaje validado**, entendido como conocimiento basado en evidencia empírica, que orienta las decisiones futuras del producto (Lean Case, s.f.).

### **Tabla 9. Fases del ciclo *build-measure-learn***

<b>Fase</b>	<b>Descripción</b>	<b>Objetivo</b>
<b><i>Build</i></b>	Construcción del MVP o experimento	Poner a prueba una hipótesis
<b><i>Measure</i></b>	Recolección de datos y métricas	Obtener evidencia del comportamiento
<b><i>Learn</i></b>	Análisis de resultados	Generar aprendizaje validado

**Fuente:** Elaboración propia con base en Lean Case (s.f.).

Este ciclo no se ejecuta de manera aislada, sino que se repite de forma continua. Cada iteración permite ajustar el producto en función del aprendizaje obtenido, configurando un proceso de desarrollo dinámico. En este sentido, el producto evoluciona a través de sucesivas versiones que incorporan mejoras basadas en evidencia.

En el marco del ciclo *build-measure-learn*, los resultados obtenidos a partir de la experimentación conducen a instancias de decisión que orientan el desarrollo del producto. Estas decisiones no se formulan de manera intuitiva, sino que se sustentan en el **análisis de evidencia empírica**, permitiendo definir el rumbo del proyecto en función del grado de validación

alcanzado. Entre las decisiones más habituales se encuentran perseverar, ajustar y pivotar, cada una asociada a un tipo de resultado y a un nivel de aprendizaje específico.

### **Perseverar** —

La decisión de perseverar implica continuar con la misma dirección estratégica del producto, en aquellos casos donde los resultados del experimento validan las hipótesis planteadas. Esto supone que existe coherencia entre el problema identificado, la propuesta de valor y la solución implementada en el MVP. Desde una perspectiva operativa, perseverar no significa mantener el producto sin cambios, sino profundizar el desarrollo en la misma línea, incorporando mejoras progresivas sobre una base que ha demostrado ser válida. En este escenario, el equipo puede avanzar hacia la ampliación de funcionalidades, la optimización de la experiencia de uso o la escalabilidad de la solución, con un mayor nivel de confianza en la pertinencia del producto.

### **Ajustar** —

La decisión de ajustar se presenta cuando los resultados muestran una validación parcial de las hipótesis, es decir, cuando algunos aspectos del producto funcionan de manera adecuada, pero otros requieren modificaciones. En este caso, el equipo introduce cambios acotados y específicos que permiten mejorar la propuesta sin alterar su lógica central. Estos ajustes pueden involucrar modificaciones en funcionalidades, mejoras en la interfaz, cambios en la comunicación del valor o redefiniciones en el recorrido del usuario. Desde el punto de vista metodológico, ajustar implica iterar sobre la base existente, incorporando el aprendizaje obtenido para optimizar el desempeño del producto en las siguientes versiones.

### **Pivotar** —

La decisión de pivotar implica una redefinición más profunda del producto, que se produce cuando los resultados indican que los supuestos principales no se validan. En este caso, el equipo reconoce que la dirección inicial no permite resolver adecuadamente el problema o no genera el valor esperado para los usuarios. Pivotar no significa abandonar el proyecto, sino reorientarlo estratégicamente, modificando elementos centrales como el segmento de usuarios, la propuesta de valor, el modelo de

negocio o la solución planteada. Este tipo de decisión requiere un análisis más amplio, ya que implica reconsiderar los fundamentos del producto y formular nuevas hipótesis a validar en ciclos posteriores.

Estas decisiones reflejan la capacidad del equipo para interpretar los resultados y actuar en consecuencia. El enfoque *lean* propone que el desarrollo de productos no se base en planes rígidos, sino en la **adaptación continua a partir de la evidencia obtenida**.

Desde una perspectiva profesional, la implementación del ciclo *build-measure-learn* permite reducir la incertidumbre y mejorar la calidad de las decisiones. En lugar de avanzar hacia una solución final sin validación, los equipos construyen conocimiento progresivamente, ajustando el producto en función de datos reales.

En síntesis, el ciclo *build-measure-learn* organiza el proceso de desarrollo en torno a la experimentación y el aprendizaje. A través de sus fases, los equipos pueden transformar hipótesis en evidencia, y esa evidencia en decisiones, consolidando un enfoque de trabajo basado en la **validación continua y la mejora progresiva del producto**.

**Estrategias para transformar el feedback del usuario en mejoras concretas sin sesgos de**

## interpretación

En el proceso de desarrollo de productos, la recolección de *feedback* representa una fuente central de información para la toma de decisiones. Sin embargo, el valor de estos datos no reside únicamente en su obtención, sino en la capacidad de los equipos para **interpretarlos de manera rigurosa y traducirlos en mejoras concretas del producto**. En este sentido, uno de los principales desafíos consiste en evitar sesgos que puedan distorsionar la lectura de los resultados y conducir a decisiones poco fundamentadas.

Una primera estrategia consiste en **vincular el *feedback* con hipótesis previamente definidas**. En lugar de analizar la información de manera general, los equipos deben interpretarla en función de las preguntas que dieron origen al experimento. Este enfoque permite evitar lecturas arbitrarias y centrar el análisis en la validación o refutación de supuestos específicos. De este modo, el *feedback* se convierte en evidencia orientada a la toma de decisiones y no en una colección de opiniones dispersas.

En segundo lugar, resulta necesario **combinar datos cualitativos y cuantitativos**. Mientras el *feedback* cuantitativo permite identificar patrones de comportamiento, el *feedback* cualitativo aporta contexto para comprender las causas de esos comportamientos. Esta integración reduce el riesgo de interpretar datos de forma aislada, ya que permite contrastar

métricas con percepciones reales de los usuarios. Por ejemplo, una baja tasa de uso puede explicarse a partir de entrevistas que revelen dificultades en la comprensión del producto.

Otra estrategia relevante es la **definición de criterios de análisis previos a la recolección de datos**. Establecer métricas, indicadores y umbrales de validación antes de ejecutar el experimento permite interpretar los resultados de manera consistente. Esto evita que el análisis se ajuste a expectativas previas o preferencias del equipo, favoreciendo una lectura más objetiva de la evidencia.

Asimismo, es importante trabajar con **muestras representativas y variadas de usuarios**, evitando basar decisiones en casos aislados o en opiniones individuales. La diversidad en la recolección de *feedback* permite identificar patrones más consistentes y reducir el impacto de experiencias particulares que no reflejan el comportamiento general.

Desde una perspectiva metodológica, también resulta clave **documentar los resultados y sistematizar el aprendizaje**. Registrar qué hipótesis se validaron, qué datos se obtuvieron y qué decisiones se tomaron permite construir un proceso acumulativo de conocimiento. Esta práctica facilita la trazabilidad de las decisiones y evita reinterpretaciones posteriores basadas en percepciones subjetivas.

Por otra parte, la implementación de instancias de análisis colectivo dentro del equipo contribuye a reducir sesgos individuales. Al discutir los resultados de manera colaborativa, se incorporan múltiples perspectivas que permiten contrastar interpretaciones y enriquecer el análisis. Esta dinámica favorece una comprensión más equilibrada de los datos.

Finalmente, transformar el *feedback* en mejoras concretas implica **traducir los hallazgos en acciones específicas de diseño o desarrollo**. Esto requiere identificar qué aspectos del producto deben modificarse, cómo se implementarán esos cambios y qué nuevas hipótesis surgirán a partir de ellos. De este modo, el *feedback* se integra en el ciclo iterativo, permitiendo que cada aprendizaje se refleje en una nueva versión del producto.

En síntesis, las estrategias para trabajar con *feedback* se orientan a garantizar que la información obtenida se utilice de manera rigurosa, evitando interpretaciones sesgadas. A través de la vinculación con hipótesis, la combinación de datos, la definición de criterios claros y la sistematización del aprendizaje, los equipos pueden transformar el *feedback* en decisiones concretas, consolidando un proceso de desarrollo basado en evidencia.

CONTINUAR

## Referencias

---

**Bassino, F.** (2016). *¿MVP es un producto o es un experimento para validar tu idea?* Medium. <https://medium.com/espanol/mvp-es-un-producto-o-es-un-experimento-para-validar-tu-idea-76ef9b76a040>

**Caneva, M.** (2022). *MVP: qué es el producto mínimo viable y cómo hacerlo.* Coderhouse. <https://www.coderhouse.com/ar/coderlibrary/mvp-que-es-el-producto-minimo-viable-y-como-hacerlo>

**EAE Business School.** (s.f.). *¿Qué es la metodología Lean Startup y cómo funciona?* <https://www.eaebarcelona.com/es/blog/que-es-metodo-lean-startup>

**Infinitia Industrial Consulting.** (2024). *Qué es MVP o producto mínimo viable y cómo realizarlo* <https://www.infinitiaresearch.com/noticias/que-es-mpv-producto-minimo-viable/>

**Lean Case.** (s.f.). *La metodología Lean Startup.*

<https://mag.leancase.co/la-metodologia-lean-startup/>

**Washington, J.** (2024). *What is a minimum viable product (MVP)?*

Amplitude. <https://amplitude.com/blog/what-is-a-minimum-viable-product-mvp>

CONTINUAR