

Estrategia de Lanzamiento y Hosting para SmartKet

Este documento complementa la arquitectura técnica (v2) y se enfoca en dos áreas críticas para el lanzamiento: cómo capturas a tus clientes (Onboarding) y dónde alojarás el sistema (Hosting).

1. El Flujo de Onboarding Perfecto (Registro vs. Pago)

Tu pregunta sobre "cuándo pedir el pago" es la más importante para un negocio SAAS.

Tu idea: Registro (Plan, RUC) -> Pago -> Setup Wizard -> Usar la App. **Problema:** Estás pidiendo dinero (el punto de mayor fricción) *antes* de que el cliente haya visto *nada* de valor. La tasa de abandono será altísima.

Propuesta: El Modelo "Free Trial" (Prueba Gratuita)

Este modelo alinea tu visión de "fácil como Facebook". Permite al cliente "engancharse" con el producto antes de pagar.

Flujo de Onboarding Recomendado:

1. Registro Rápido (en `smartket-landing`):

- **Lo mínimo:** Email, Contraseña, Nombre del Negocio.
- (Opcional) Un campo: "¿Cuál es tu rubro?" -> [Pollería].
- **¡No pidas plan, RUC ni tarjeta de crédito aquí!**

2. Inicio de Sesión Automático y Redirección:

- Inmediatamente después del registro, el sistema lo loguea y lo redirige a `app.smartket.com`.
- Tu API (`smartket-api`) ya le creó su BBDD aislada (`db_cliente_123`) y le asignó el plan `TRIAL_POLLERIA` en la `admin_db`.

3. El "Setup Wizard" (¡Ahora!):

- **¡Aquí es donde pides el RUC!** Es el primer paso del wizard, no del registro.
- **Paso 1: Datos Fiscales.** "Hola Juan, configuremos tu pollería."
 - RUC (Aquí llamas a tu API para validar el RUC. Si no tiene, puede marcar un check "Vender con Boleta Simple").
 - Nombre Legal, Dirección, Moneda (Soles).
- **Paso 2: Sucursal.** "Crea tu primer local."
- **Paso 3: Caja.** "Configura tu primera caja."
- **Paso 4: Importar Productos.** (Este es tu punto 3).
 - [Añadir producto manualmente]
 - [Descargar plantilla de Excel para importación masiva]
 - [Subir Excel con mis productos] (¡Esto es un salvavidas para tus clientes!)

4. Uso de la App (El "Aha! Moment"):

- Terminado el wizard, el cliente entra al dashboard. ¡Ya está todo configurado con *sus* datos!

- Puede agregar un producto, hacer una venta de prueba, imprimir un ticket. Ha visto el valor.

5. La Conversión (El Pago):

- Tu plan `TRIAL_POLLERIA` tiene un límite (ej. 14 días gratis, o "hasta 100 ventas").
- Durante el trial, un banner sutil dice "Te quedan 13 días de prueba".
- Cuando el trial termina (o alcanza el límite), la app se bloquea amigablemente: "Tu prueba ha terminado. ¡Elige un plan para continuar donde lo dejaste!"
- **Ahora el cliente SÍ pagará**, porque ya configuró todo su negocio y vio que el sistema funciona.

Respuesta a tu pregunta: El **Setup Wizard** es lo *primero* que el usuario debe ver *después* de registrarse, y debe ser *antes* de pagar. Es el corazón de tu prueba gratuita.

2. El Hosting: Hostinger vs. "La Nube"

Tu pregunta sobre Hostinger es excelente y muy técnica. Aquí la respuesta es crucial.

Debes diferenciar "Shared Hosting" (Hosting Compartido) de "VPS" (Servidor Privado Virtual).

A. Shared Hosting (Hosting Web Barato)

- **¿Qué es?** Son los planes de "Hosting Web" de Hostinger, GoDaddy, etc. Cientos de webs viven en *un solo* servidor.
- **¿Funcionará para SmartKet? NO. Rotundamente NO.**
- **Por qué no:**
 1. **Rendimiento:** Tu arquitectura de BBDD aislada no sirve de nada si todas esas BBDD viven en un servidor lento y compartido. Un cliente con muchas ventas (el "vecino ruidoso") hará que el sistema de *todos* tus clientes sea lento.
 2. **Permisos: No podrás crear bases de datos dinámicamente.** Los hostings compartidos no te dan permisos de `CREATE DATABASE` sobre la marcha.
 3. **Scripts (Backups/Migraciones):** No tendrás control total para ejecutar tus scripts de `cron jobs` (tareas programadas). Estarás muy limitado.

B. VPS (Servidor Privado Virtual)

- **¿Qué es?** Es la solución correcta. Es un "servidor en la nube" para ti solo. Tienes control total (acceso `root` por SSH).
- **¿Hostinger lo tiene?** Sí. Hostinger (y todos los proveedores: DigitalOcean, Vultr, AWS, Google Cloud) venden planes **VPS**.
- **¿Funcionará para SmartKet? SÍ. Esta es la solución.**

Conclusión del Hosting: No importa si eliges Hostinger, DigitalOcean o AWS, siempre y cuando contrates un plan **VPS**, no un "Hosting Web" compartido. Un VPS te da el control total que tu arquitectura *multi-tenant* necesita.

3. Tus Scripts de Backup en un VPS

Ahora que sabemos que usarás un VPS, tu pregunta sobre los backups tiene respuesta:

Sí, tu plan de scripts es perfecto para un VPS.

1. ¿Cómo funciona?

- En tu VPS (que es un servidor Linux), configuras un `cron job` (una tarea programada).
- Le dices: "Ejecuta este script todos los días a las 3:00 AM".

2. ¿Qué hace el script? (Puede ser un script `bash` o un Comando de Laravel).

1. Se conecta a tu `smartket_admin_db`.
 2. Hace `SELECT nombre_db FROM tenants WHERE status = 'ACTIVO'`.
 3. Recorre esa lista (un `loop`).
 4. Para cada nombre de BBDD (`db_cliente_123` , `db_cliente_124` ...):
 - Ejecuta `mysqldump db_cliente_123 > backup-123.sql.gz`
 5. (Opcional pero recomendado) Sube esos archivos `*.gz` a un almacenamiento externo (como Amazon S3 o Backblaze B2) para que los backups no estén en el mismo servidor que las bases de datos.
3. **Migraciones:** Lo mismo aplica para las migraciones. Creas un comando de Laravel que lee la lista de tenants y ejecuta las migraciones en cada BBDD.

Resumen: Tu estrategia es sólida. Enfócate en el **MVP de Pollería**, usa el **Flujo de Free Trial** para captar clientes, y despliega tu aplicación en un **servidor VPS** (de Hostinger o cualquier otro) para tener el control que tu arquitectura necesita.