

FROM MY JOURNEY

THE MACHINE WAS READY. NOW I HAD TO TURN ON THE TRAFFIC.

I had the product, sales page, checkout, payment methods, automated delivery, business email, custom domain and a working buyer path. Then I hit the next wall: building the store does not put people inside it.

My first instinct was to jump into ads. Instead, the live setup forced me to learn one more rule: **traffic comes after the machine can receive a YES**. Tracking, payment, delivery and the actual customer ZIP all have to be ready before strangers arrive.

BUILD THE MACHINE → TEST THE MACHINE → TURN ON TRAFFIC

The lesson

Do not confuse creating an ad with being ready to advertise. An ad is almost the last switch. Before you flip it, make sure the business behind the click can identify the visit, accept the payment, deliver the product and help the customer.

This section shows the route I actually used. Your platform may look different. The principle is the same.

1. BUILD YOUR AD IDENTITY

Before creating the ad, I separated my personal Facebook login from the business assets. Meta uses several layers, and they can look like the same thing when you are new.

PERSONAL LOGIN → BUSINESS PORTFOLIO → FACEBOOK PAGE → AD ACCOUNT → DATASET / PIXEL

My working setup

- Business Portfolio created for the business.
- Facebook Page connected to the portfolio.
- Dedicated ad account created with the correct country, currency and time zone.
- Billing method attached and visibly confirmed after refresh.
- A new business-owned dataset / Pixel created instead of reusing an old unrelated asset.

Wall I hit

Meta showed old accounts and assets in different menus. That did not mean they belonged to the new business. I learned to verify ownership and the exact ID instead of trusting the name alone.

Definition of done: you know which Page, ad account and dataset belong to this launch, and you can identify them without guessing.

2. CONNECT THE WEBSITE TO META

I used Systeme.io for the funnel. I first connected the Meta base Pixel globally so every funnel page could send a PageView. Then I connected Systeme.io's server-side Conversions API using the Pixel/Dataset ID and access token.

Important: the access token is a secret. Do not publish it, screenshot it for customers or paste it into support chats.

BASE PIXEL = PAGEVIEW | PAGE EVENT = WHAT HAPPENED | CAPI = SERVER COPY

What I verified

- The live page source contained the correct Pixel ID.
- The browser loaded Meta's fbevents.js and sent tracking requests.
- Events Manager received browser events.
- Events Manager later showed server / Conversions API events too.
- I learned that Meta reporting can lag, so I verified execution before rebuilding a working setup.

Do not stop at 'Saved.'

A setting can be saved and still not be live where you expect. Open the real public page and verify the Pixel is actually present. This single check saved me from diagnosing the wrong problem.

3. TRACK THE FUNNEL, NOT JUST THE DOMAIN

The global base Pixel told Meta that a page loaded, but it did not automatically tell Meta what that page meant. The individual funnel pages needed their own standard events.

SALES PAGE → ViewContent CHECKOUT → ViewContent THANK YOU → Purchase

The two-layer setup

Browser event: page-specific Meta event code runs in the page header.

Server event: the Systeme.io Facebook-event dropdown sends the corresponding CAPI event.

For my checkout page, the base Pixel was already present globally, but searching the live source for **ViewContent** returned zero. That proved the dropdown alone was not creating the browser ViewContent event. I added the page-specific event code and verified it appeared in the live source.

For the Thank You page, I used Purchase because that page represents the completed buyer path. Meta's own standard-event guidance recognizes a thank-you or confirmation page as a normal place to fire Purchase.

Why Event IDs matter

When the browser and server both report the same action, matching event IDs allow Meta to treat them as the same event rather than two separate conversions. Use the current placeholders/instructions supplied by your funnel platform; platform syntax can change.

4. BUILD THE FIRST AD AS A TEST

I did not build ten campaigns. I built one controlled first sales test so I could learn from clean data.

**ONE OFFER → ONE DESTINATION → ONE MAIN CREATIVE → SMALL TEST →
READ THE NUMBERS**

What I set up

- Sales campaign with Website as the conversion location.
- One ad set for the first market test.
- One dedicated ad creative matched to the Business Planner brand.
- Destination pointed directly to the secure sales-page URL.
- Square, vertical and horizontal crops reviewed one at a time.
- Automatic creative options reviewed instead of accepting every default blindly.
- Ad left in Draft while the machine and customer package were finished.

Wall I hit

Meta can auto-crop a good image into a bad ad. My headline and product mockup were cut in some placements. The fix was simple: review each crop separately. Automation is useful, but the final ad still needs human eyes.

Controlled does not mean perfect.

The first ad is there to produce evidence. Keep the variables understandable so you can diagnose what happens after the click.

5. READ THE FUNNEL LIKE A DOCTOR

Once traffic starts, do not judge the business from one number. Follow the customer through the machine.

IMPRESSION → CLICK → SALES PAGE → CHECKOUT → PURCHASE → DELIVERY

Simple diagnosis

Signal	What to inspect
Impressions but few clicks	The ad/hook/creative may not be earning attention.
Clicks but weak sales-page movement	The promise, page, audience or message may not connect.
Sales-page interest but weak checkout movement	Offer clarity, trust, price or CTA may need work.
Checkout activity but no purchases	Check payment friction, trust, errors and offer objections.
Purchase but no delivery	This is an operations failure. Fix it immediately.

Do not kill or celebrate the whole business from a tiny amount of data. The first job is to find where the machine is leaking and what evidence you need next.

6. BEFORE YOU TURN ON TRAFFIC

This is the final launch gate I wish I had before I started touching ads.

- **REAL PRODUCT:** the ZIP contains the actual customer-ready V1 - not a demo or placeholder.
- **PRICE:** sales page, checkout, product and delivery email all show the intended offer and price.
- **SALES PAGE:** every CTA goes to the correct checkout.
- **CHECKOUT:** card/payment methods work on the current custom domain.
- **THANK YOU:** successful buyers land on the correct confirmation page.
- **DELIVERY:** New Sale automation sends the correct ZIP and support information.
- **EMAIL:** sender name/address are correct; SPF, DKIM and DMARC have been checked.
- **DOMAIN:** the exact URL used in the ad loads securely on desktop and mobile.
- **TRACKING:** correct Pixel/Dataset is on the live site; PageView and page-specific events are present.
- **PURCHASE TRACKING:** Thank You is configured as Purchase and the event is verified in Meta when available.
- **AD DESTINATION:** the ad points to the sales page, not checkout, tester hub or an old domain.
- **AD CREATIVE:** text is readable in square, vertical and horizontal placements.
- **AD COPY:** promise matches the product; no guarantee or claim the product cannot support.
- **BILLING:** ad-account payment method is attached and confirmed.
- **SUPPORT:** support@100ksolution.com is monitored and the customer knows where to ask for help.
- **REFUND/TERMS:** customer-facing refund/terms/privacy information is present where required for the launch.
- **MOBILE:** sales page → checkout → confirmation → email/download has been checked on a phone.
- **SELF-TEST:** you have walked the exact buyer journey yourself.
- **MEASUREMENT:** you know which numbers you will record after traffic starts.

ALL TRUE? NOW THE AD CAN BRING PEOPLE INTO A MACHINE THAT IS READY.

NEXT: THE REPS

Traffic is not the finish line. Once real people enter, SELL → CREATE → IMPROVE becomes the repetition loop. Document what happens, fix the biggest leak, and keep moving.