For confidentiality reasons, the client's name is not disclosed. The client is a renewable energy aggregator and a major player in France's energy transition. Some features are intentionally not described, as they are strategically sensitive.

How a renewable energy aggregator automated its commercial operations in 10 months using low-code

Summary

A renewable energy aggregator, a major player in France's energy transition, wanted to speed up the commissioning of solar projects by simplifying and securing its exchanges with its solar park development partners (such as EDF Solutions Solaires).

The problem: a business-critical process still relied on Excel files, manual checks across multiple departments, and complicated contracts to fill out. At the same time, the internal IT teams had neither the bandwidth nor the short-term priority to launch a "traditional" project.

In 10 months, working as an external low-code consultant and product builder, I:

- Reviewed the stack to adapt it to the actual use case on the intermediary side
- Automated operations from start to finish (from data collection to DocuSign signature, including numerous technical and administrative checks)
- Built a robust v1, with 250+ "functional recipes" (business rules and scenarios)
- Trained an internal developer to ensure continuity and future developments

Result: the client now has an operational platform that generates a price quote and a pre-filled contract in seconds, where teams previously had to go through a series of Excel manipulations and scattered manual checks.

In short

- **Client**: a renewable energy aggregator, subsidiary of a major French energy group
- **Target**: Solar park development partners
- **Before**: Excel, multi-departmental manual checks, redundant contracts
- **Initial stack**: n8n, ClickFunnels, DocuSign, Salesforce
- **Target stack**: n8n, WeWeb, Xano, Auth0, DocuSign, Salesforce
- **Duration v1**: 10 months, without dedicated bandwidth on the internal tech teams' side (3 months of discovery and POC)
- **Keys to success**: low-code product building, high business density (250+ rules), near-total automation of operations

1. Context and challenges

This renewable energy aggregator plays a central role in promoting renewable energy. To grow, the company relies on a network of solar park developer partners: these intermediaries identify, set up and finance projects, then work with this aggregator to promote the energy produced.

For these partners, time and operational clarity are essential:

- Acting quickly to secure the best land opportunities
- Immediately understand the technical and economic conditions
- Limit administrative and contractual back-and-forth

However, in the context of industrialisation, the initial process no longer met these requirements.

2. The starting point: Excel, manual checks, redundancies

When I arrived, the process resembled what many organisations still experience today:

- **Information collected via Excel files** sent and returned by e-mail
- **Manual checks** in several departments (technical, commercial, legal, etc.)
- **No clear ownership** of certain checks, meaning that several people were repeating the same checks
- **Long and redundant contracts**, with up to eight repetitions of the same information
- **Appendices sent after the fact**, creating additional friction on the client side

This system comes at a price:

- **Structural slowness**
- **Risk of error** (copy/paste, file versioning)
- A **high operational burden** for teams.

Furthermore, the first stack chosen before my arrival—**n8n, ClickFunnels, DocuSign, Salesforce**—had been designed for a different use case, closer to **"traditional" B2B** (campaigns, landing pages, simple conversion funnels). It had performed perfectly during the POC phase, but reached its limits as soon as the business complexity and industrialisation requirements increased.

3. Why a low-code project rather than a "traditional" IT project?

Two major constraints guide the strategy:

1. **Lack of availability of internal tech teams**
    - The IT and Salesforce teams already had a busy roadmap
    - It was not realistic to add a new complex project to their short-term priorities
2. **Time-to-market challenge**
    - The solar project market is evolving rapidly
    - Postponing this project for one or two years to wait for IT bandwidth would have created a real opportunity cost

In this context, a **traditional IT project** (specifications >> development >> testing >> deployment cycle) would have been:

- Slower to get started
- More costly in terms of internal mobilisation
- Risky in terms of the gap between business needs and final delivery

Low-code product building allows for a different approach:

- **Rely on proven building blocks** (Auth0, Xano, WeWeb, n8n, DocuSign, Salesforce)
- **Quickly build a "real" business application**, not just forms
- **Iterate on the fly with operational staff**, without monopolising internal tech teams
- Maintain **high standards for robustness** (security, rights, traceability).

4. Rethinking the stack for solar farm developer partners

4.1. From the initial stack to a "business platform" architecture

Before I arrived:

- n8n, ClickFunnels, DocuSign, Salesforce

- A very relevant set for a marketing/lead generation funnel, less so for a complex operating application.

After my intervention:

- **n8n**: orchestration and automation engine
- **WeWeb**: low-code, user-friendly and rapidly iterable web front-end
- **Xano**: no-code backend/API, robust and adapted to complex business rules
- **Auth0**: identity and access management
- **DocuSign**: electronic signature
- **Salesforce**: still at the heart of the CRM ecosystem

Why this change?

- **Aligning the stack with the right use case**: a platform used on a daily basis by **solar project operators**, not just a simple external form
- **Increasing robustness**: scalability, fine-grained rights management, complex validations, logs, error recovery, etc.
- **Limiting the impact on the IT department**: the stack interfaces cleanly with existing systems, without requiring them to be completely rebuilt

The goal was not to "throw away" the previous work, but to **capitalise on what was working** (n8n, DocuSign, Salesforce) by adding the missing pieces to move from POC to an **industrial platform**.

5. Automate end-to-end operations

The ambition agreed upon was clear: **to design a process that leaves virtually no room for human intervention**, with the exception of the **DocuSign signature**.

5.1. 250+ "functional recipes": the business encoded in the platform

From v1 onwards, we integrated **more than 250 functional recipes**:

- Technical and operational rules for **individual fields**
- Rules on **field combinations** (e.g. consistency between financing type, site characteristics, injection profile, etc.)
- Management of **multiple financing types** and associated **commercial logic**
- Validation of data eligibility, consistency and completeness

- Specific scenarios depending on the type of project or partner

In practice, this means that much of the expertise of the company's teams is now **encapsulated in the tool**: errors that would be possible in Excel are simply **not allowed** in the platform.

5.2. Automatic checks via APIs and internal logic

To make the process even more reliable, we have connected the platform to:

- **Specialised APIs** (such as Agence ORE, Géoservices, etc.) to validate certain project data or technical parameters
- **Internal rules coded in Xano**, which reproduce recurring business decisions

In concrete terms:

- The tool automatically queries the right sources
- Compares the data entered with business constraints
- Rejects or adapts the options offered to the end user.

Teams no longer "check"—they manage.

5.3. A commercial offer generated in seconds

One of the major business pain points concerned the generation of offers:

Before:

- Data extraction
- Multiple Excel manipulations
- Research and cross-checking of information
- Risk of errors in prices or assumptions.

After:

Using **a postal address** or **GPS coordinates**:

1. The platform retrieves the necessary information (via APIs and internal rules)
2. Applies the correct commercial and technical rules
3. Automatically calculates the relevant parameters

4. Generates a **commercial offer with shared pricing** in **a matter of seconds**

Sales representatives move from **manual production** work to **explanation and advice**, which is exactly where they are most valuable.

5.4. A pre-filled contract, ready to be signed

Contractualisation:

Before:

- Long contracts, with up to **8 instances of redundant data entry** for the same information
- Appendices sent after signing (two additional documents)
- Risk of human error, omissions, inconsistent fields

After:

- The platform generates a **fully pre-filled contract** based on data that has already been collected or extracted and validated
- The customer receives a document **ready to be signed in DocuSign**
- The two appendices that were previously sent afterwards are now **integrated into the main contract**:
    - Time savings for the customer
    - Less administrative friction,
    - The process is perceived as simpler, with no additional workload for internal teams.

6. Results: agility, delivery and total automation of operations

6.1. A v1 in 10 months despite zero IT bandwidth

In **10 months**, we have:

- Redefined the stack
- Modelled the business process in detail
- Implemented more than **250 business rules** in the platform

- Delivered an **operational v1** for partners
- Trained an internal profile to ensure continuity

All this in a context where the **internal tech teams did not have** the **availability** or capacity to prioritise this project themselves.

In a traditional model, it is likely that the project would have been:

- Postponed for several roadmap cycles
- Or launched later with fixed specifications, at the risk of being out of step with the reality on the ground

6.2. Near-total automation of operations

The new system enables:

- **End-to-end automation** of technical and operational checks
- A **drastic reduction in Excel manipulations**
- A **reduction in double (and triple) checks** between departments
- **Data security** (rights, authentication, traceability via Auth0, Xano, Salesforce)

Teams can now focus on:

- Optimising the offering
- Partner relations

6.3. Improved experience for partners

For **solar park developers**, this changes their daily routine:

- A smoother and faster process
- Offers generated almost instantly
- A clear, pre-filled contract, without endless re-entering of data
- Fewer email exchanges to correct omissions or inconsistencies

A better partner experience translates directly into:

- A more professional image
- Greater confidence in the data

- An increased ability to handle more projects simultaneously

6.4. Transmission and internal autonomy

From v1 onwards, I have:

- Structured the platform so that it could be **maintained by an internal profile**
- **Trained an internal developer** on the stack (WeWeb, Xano, n8n, Auth0)
- Implemented best practices (organisation of rules, nomenclature, documentation)

The goal was not to create dependency, but rather to make the company **autonomous in terms of future developments**.

7. What this case study says about a low-code/AI project vs. a traditional project

Even though this project is not (yet) based on generative AI, it illustrates very well the logic of a **modern low-code & intelligent automation project**:

1. **First, the business and rules are encoded** in a robust platform (in this case, 250+ functional recipes).
2. **Repetitive decisions are automated** via APIs and internal logic
3. We lay the groundwork for **later connecting AI building blocks** (e.g., project scoring, partner assistants, automatic document analysis, etc.).

Compared to a traditional project:

- **Reduce time to market.**
- **Limit pressure on IT teams**
- **Maintain high standards** of quality, security and robustness
- **Capitalise on existing tools** (Salesforce, DocuSign) instead of starting from scratch

8. And for your company?

If you recognise yourself in any of these situations:

- Your operations still rely on **Excel + emails + manual checks**
- Your IT teams are **overwhelmed** and cannot prioritise your project

- Your business depends on **partners or intermediaries** who expect speed and clarity
- You want to pave the way for **low-code & AI** applications without rebuilding everything

then support from a **low-code Consultant** and **Product Builder** can enable you to:

- Deliver a v1 in weeks or months, not years
- Automate your operations reliably
- Retain your internal teams while building their skills
- Lay a solid foundation for your future AI projects