

ENTERPRISE AI · CASE STORY

Three AI engines, three separate homes

The AI worked. People just couldn't find it. Designing one home for three AI features.

4 min

Click or press → to continue

Imagine a store whose three best departments have no signs.

One is behind a door, one is up a back staircase, one only opens if you know the knock.

WebFOCUS had three AI engines: natural-language questions, automated insights, and machine learning.



Ask a question and **Insights**: buried several clicks deep in menus, then inside a popup.



Machine learning: reached by right-clicking a dataset, an entry added in the earlier ML revamp. Before that, it was hidden inside Reporting Server.

Two paths, two mental models.

Powerful. Paid for. Impossible to find.

Rule 1

Diagnose before you solve.

THE FIRST READ

The models need work.

THE REAL PROBLEM

People couldn't find them.

The engines weren't the problem. Their addresses were.

Rule 2

Architect before the tickets exist.

The workflows across all three engines were designed first. The tickets were written after the mockups.

Rule 3

Integrate. Don't rebuild.

3 engines



1 home

The engines stayed exactly as they were. Only their addresses changed.

Rule 4

The dataset is the project.

TOOL-CENTRIC

Pick the data four times.

Every tool remembers its own dataset.

PROJECT-CENTRIC

Pick it once.

The dataset binds every tab. A new dataset opens a new project.

Rule 5

Outcome first. Tool names never.

Describe your goal. The hub routes you to the right engine.

What do you want to find out?

And a "What are...?" explainer that takes the fear out of AI.

● PRINCIPLE

Recognition over recall

Don't make people remember which tool does what. Show them the outcome they can choose.¹

Rule 6

Win the room with evidence, not taste.

List views were the exact reason these features were invisible. Large tiles that describe outcomes, backed by industry precedent and interaction logic, won the argument.

Newest in a room of architects who'd been there 20 to 30 years. Won it with evidence.

Rule 7

Iterate until it lands.

- 1 Too dense
- 2 Too passive
- 3 Competed with the main navigation
- 4 Landed

WHERE IT STANDS

Built. Not shipped.

- ✓ Three menus into one hub
- ✓ Zero new capabilities, just findable ones
- ✓ Four iterations to land the page

Engineering had built 80–90% of it when layoffs and budget cuts shelved the release, and priorities shifted toward retaining existing clients over attracting new ones. These are design outcomes, not shipped metrics.

BEHIND THE STORY

I'd already designed each engine on its own.

I had designed and modernized natural-language questions, Insights and the ML workflow one by one. Each sat in its own menu, each barely found.

So I pitched one home for all three, and owned it end to end.

Not more features. One architecture.

CHEAT SHEET

Making AI features findable

- 1 **Diagnose first.** Low use isn't always a model problem.
- 2 **Architect before tickets.**
- 3 **Integrate, don't rebuild.**
- 4 **The data is the project.** Pick it once.
- 5 **Outcomes, not tool names.**
- 6 **Win with evidence.**
- 7 **Iterate until it lands.**

READ MORE

The full case study.

The routing fight, the four iterations, and the architecture.

Full case study: anujaharsha.com/work/iq-plugin

Enterprise story 03: "Three AI engines, three separate homes." Designed by Anuja Nimmagadda for WebFOCUS at Cloud Software Group.