

CASE STUDY

COP

Installed Base

+ Sites

Hitachi Energy — Collaborative Operations Portal

Designing a scalable enterprise portal experience across multiple business units — from fragmented discovery through card sorting, lo-fi, UI direction, and post-MVP evolution.

Solo UX/UI designer — end to end responsibility

From discovery and facilitation to final UI decisions and post-launch iteration.

TIMELINE

~6 months discovery → finalized UI
+ post-MVP iteration

METHODS

Workshops · Proto-personas · Card
sorting
Lo-fi wireframes · UI direction

STAKEHOLDERS

Process owner · BA · 6+ business units
Internal + external users

PLATFORM

Enterprise portal · Internal & external
access · Multi-BU scope

What COP was — and why it mattered.

An enterprise portal, not an isolated feature

COP (Collaborative Operations Portal) served both external users via the Hitachi Energy website and internal users through company applications. When I joined, it was still largely centred on Grid Automation — but was already expected to scale into a multi-BU enterprise platform, eventually replacing an older customer portal under the broader TSA / Customer Connect / ex-MyABB direction.

I was the only UX/UI designer fully dedicated to this project

That is not a detail — it defines the nature of the work. I was not polishing pre-defined flows or decorating finished requirements. I was carrying UX/UI responsibility end to end: creating clarity where ambiguity existed, translating stakeholder input into design direction, structuring the experience, and making the detailed interface decisions that determined how the product actually worked.

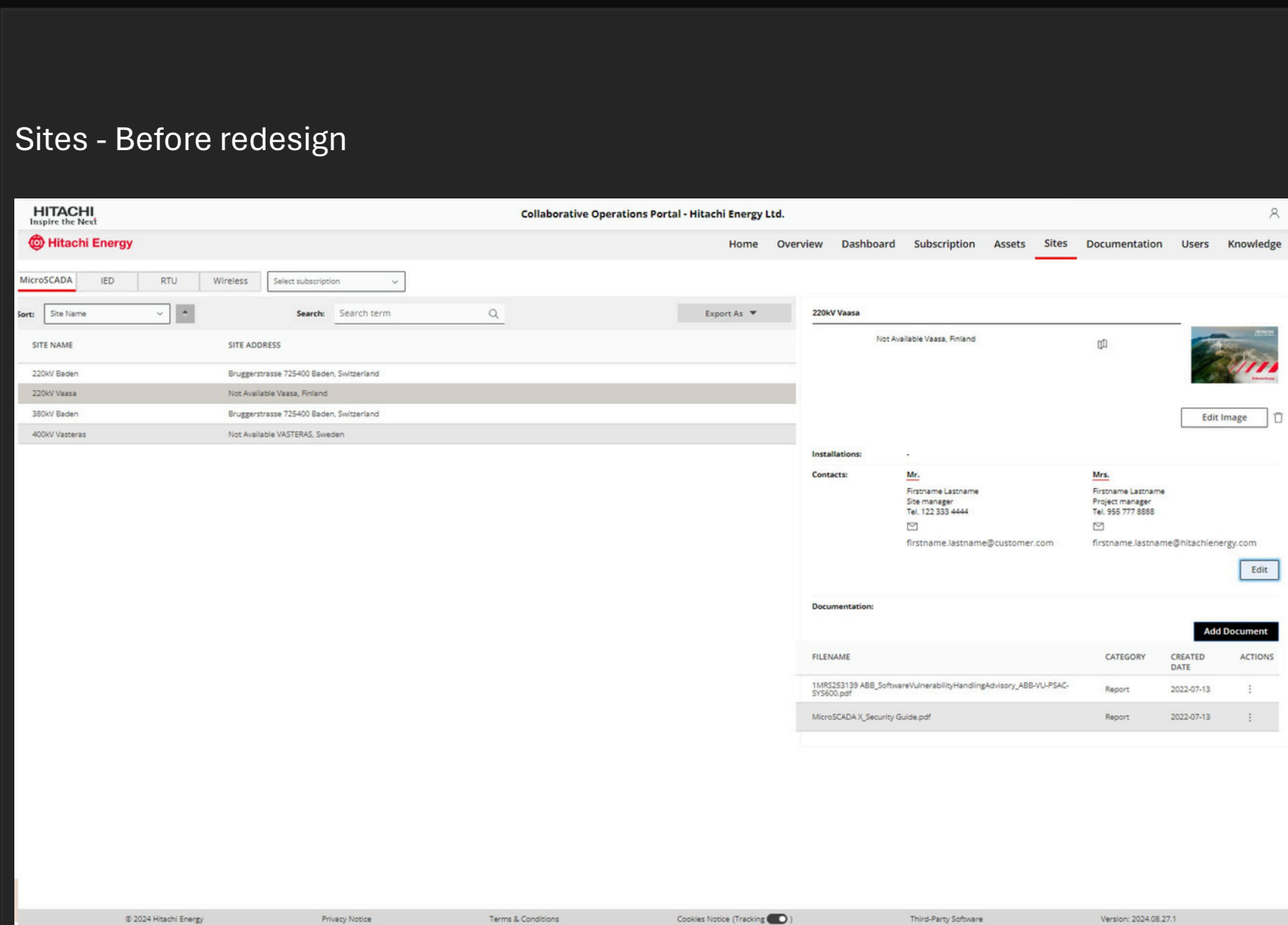
Broader context shaped every local decision

The longer-term COP roadmap included case management, SLA visibility, self-service, e-commerce, and lifecycle support across BUs. Even when designing an asset page, I was designing inside a portal trying to evolve from a legacy tool into a scalable platform. That awareness shaped how I approached scope, restraint, and future-proofing throughout the project.

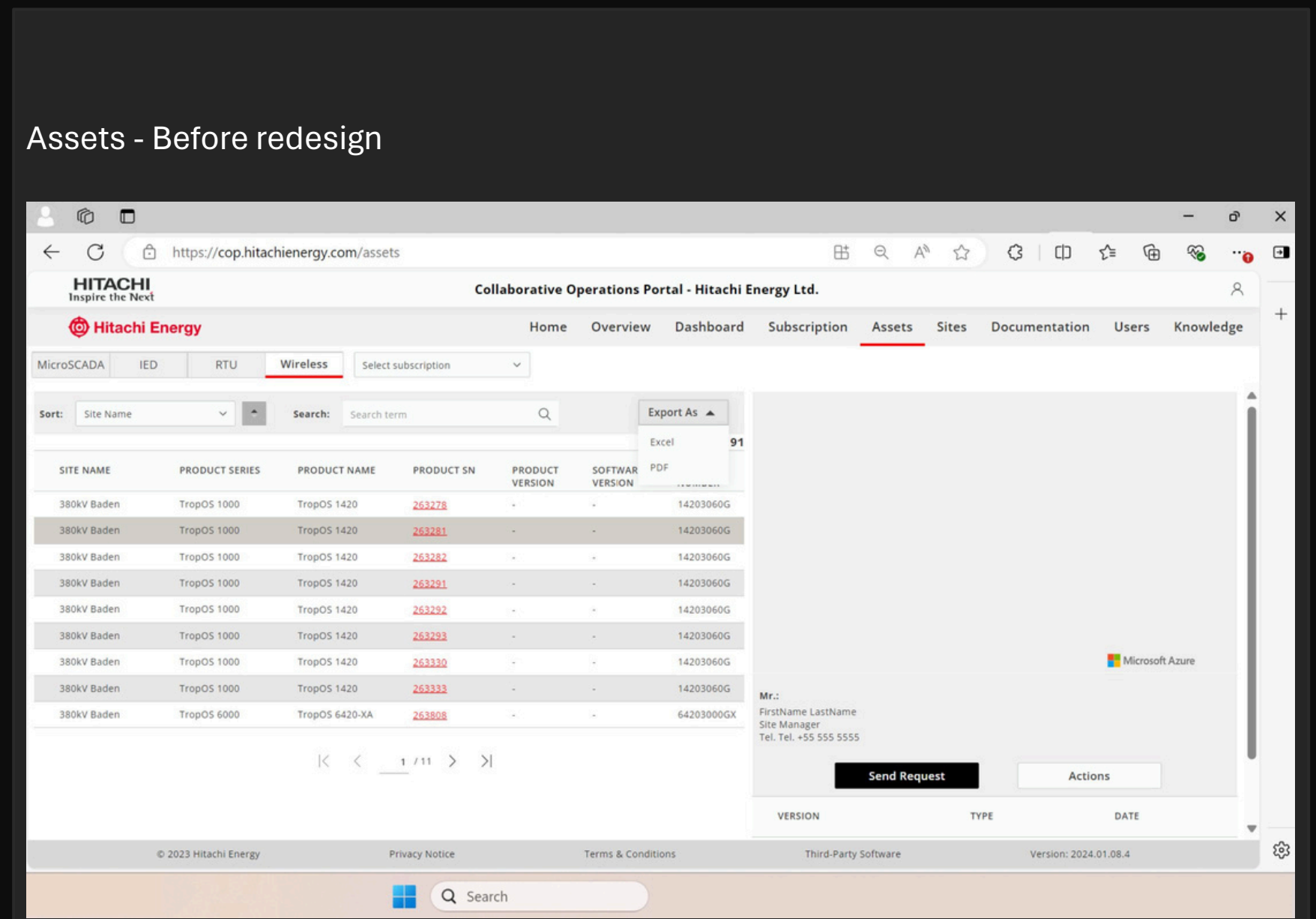
The starting state: limited, single-BU, no clear user model.

There had been portal-level user interviews around 2022. I reviewed them carefully — but also critically. They were broad, they were done years earlier, and the portal had evolved significantly. I extracted recurring themes as background context, but did not treat them as reliable current design input. That critical distance from existing research is itself a design decision.

SITES — Flat list + side panel (Grid Automation only)



ASSETS — Tab navigation, no hierarchy, no product catalog



This was not a screen redesign. It was a structure problem.

Key problem

How do you design one Installed Base experience for 6+ business units without turning Assets into an overloaded catch-all page — while also starting a portal-wide visual rebrand at the same time?

No shared user model

Portal served internal and external audiences, but no current, shared understanding of who those users were.

6+ BUs, competing views

Each business unit had different products, terminology, and expectations for the same functionality.

Legacy pages insufficient

Old Grid Automation Assets and Sites existed — but were limited and not scalable to the broader direction.

Limited user access

Direct customer contact was restricted early on. Most initial input came through stakeholder interpretation.

Scope kept expanding

Assets started as a visibility page and quickly absorbed lifecycle, documentation, service history, and more.

Parallel rebrand

The PO wanted the portal's visual language modernised at the same time as the Installed Base work was progressing.

Broad discovery first. Then structure. Then UI.



01

Workshops & Proto-Personas

Led cross-BU sessions with the process owner. Asked BU reps to describe both internal and external audience examples. Used outputs as working models — not formal validated personas, but genuinely useful for surfacing shared needs across BUs.



02

Field Mapping & Data Context

Connected workshop input to actual field-level requirements: what site and asset data would need to surface in the portal. Also connected to Salesforce-side findings from the BA to bridge abstract stakeholder wishes with real product structure.



03

Low-Fi Wireframes

Started simple and structural — map-based Sites overview, catalog-first Assets entry point. Kept polish deliberately low to turn an ambiguous conversation into something visible and improvable. Iterated every 1–2 weeks with stakeholders.



04

Hybrid Card Sorting

Initiated this myself as the scope broadened. Prepared cards for current, future, and internal-only functions. Asked participants to group without anchoring to legacy structure. Results exposed the Assets overload risk before it became a design problem.



05

UI Direction & Rebrand

Evolved lo-fi into finalized UI direction while simultaneously contributing to the portal's visual refresh. Made deliberate decisions around hierarchy, button prominence, column order, information sequencing, and controlled use of red as an emphasis signal.

Stakeholder reviews: weekly or every two weeks throughout. I was the only UX/UI designer carrying this work across the full span.

I initiated this exercise myself. It was not in the original brief.

Cards covered current portal items, future items, and internal-only functions. Participants were encouraged to add their own stickers. The key rule: group by meaning, not by how the current portal is structured. This prevented legacy bias from contaminating the mental models.

The screenshot shows a digital workspace for a hybrid card sorting exercise. At the top, a purple header reads "Transformers (TR) Transformers Service". Below it, a grid of cards is organized into three main sections: "Current items", "Internal items", and "Future items".

- Current items:** A grid of 24 white cards with rounded corners, arranged in three rows and eight columns. The cards contain various service-related terms such as "Applications available to you", "Complementary Services", "e-Business Platform", "Return of Reflected Authorization", "API Documentation", "Local API Documentation", "FAQ", "Check your assets", "Application users", "Select subscription", "General subscription information", "(group) License/Status", "(group) Next Operating System", "(group) Internal Product versions", "Current Subscription", "Webinars / Trainings", "Downloads (Articles)", "Product Catalog", "Site of Sites Location", "Release notes", "Table of Sites", "Table of Assets", "Documentation releases", "Manuals and Product guides", and "Certificates and Reports".
- Internal items:** A grid of 6 green cards with rounded corners, arranged in three rows and two columns. The cards contain: "Table of users", "Add user", "Refresh users", "Full Subscription", "Select subscription", and "For Subscription".
- Future items:** A grid of 12 yellow cards with rounded corners, arranged in three rows and four columns. The cards contain: "Recent update", "New card for a specific user", "Asset Specific Documentation", "Table with Case History & Overview", "Recommended Products", "Recent update", "Self-service functionality for a list of services (e.g. maintenance)", "Asset Service History", "Notifications", "Asset registration/update", "New or new products releases", "Table with SLA service", "Product Specifications", and "Subscription/Case Plan of SLA".

To the right of the cards, there is a section titled "If you think something was missing, feel free to add new items (Optional):" with three blue cards, each containing the text "I'm sure".

At the bottom left, there is a large, multi-colored sticky note area with four main sections: "Homepage" (pink), "Overview" (yellow), "Dashboard" (green), and "Subscription" (blue). To the right of these sections are three vertical grey bars with yellow labels: "I am not sure", "Unclear", and "Redundant".

In the bottom right corner, there is a zoom control bar showing a magnifying glass icon, a minus sign, the text "28%", and a plus sign.

Card Sorting Results — and a structural risk revealed.

ASSETS

Wide — and at risk of overload

12+ items were placed on Assets: documentation, service history, case actions, lifecycle status, product specs, upgrade flows, spare parts, registration, SLA expectations. The page risked becoming a catch-all bucket. Surfacing this risk early was itself a UX outcome — it protected the product from a poor IA decision before it was built.

Card Sorting in terms of Assets and Sites:



- Check your Assets: **8**
- Assets Specific Documentation: **7.5 - Future item? (kind of exists already)**
- Raise Case for a Specific Asset: 7 - Future item**
- Asset Service History: **6 - Future item? (kind of in new design - table item)**
- Asset registration / update: 6 - Related to future item. Repeating as below.**
- Product Specifications: **6 - Future item? (kind of in new design - page)**
- Product Catalog: **4 - On the new design (but maybe the name could be more clear)**
- Return of Material Autorization: **4 - Currently it is on the Homepage**
- Request Update: 4 - Future item**
- Request Upgrade: 4 - Future item**
- Certificates and Report: **4 - Currently in "Documentation" nav item**
- Recommended Products: **3 - Future item**
- Table with SLA Overview: 0 - Future item (kind of in new design - table item)**
- Select Subscription: **0**

SITES

Clear — geography does the work

Sites had only two strong items: a map and a list of sites. Geography gave participants an intuitive structure with no ambiguity about where things belonged. No overload risk. This contrast — Sites' clarity vs Assets' breadth — became the central structural insight of the whole project.

Card Sorting in terms of Assets and Sites:



- Map of Sites: **6 - Exists in new design**
- Table of Sites: **6**

Select Subscription: **0**
 Anything else that appeared had only 1 point and was not related to sites.

The Final Analysis will be on Miro (UX UI file)
 including analysis based on the other navigation items

- We should take into account that Asset page should not be overloaded as well to not confuse the user as a lot of items were placed on Assets Page during the exercise.

Structure first. Polish second.

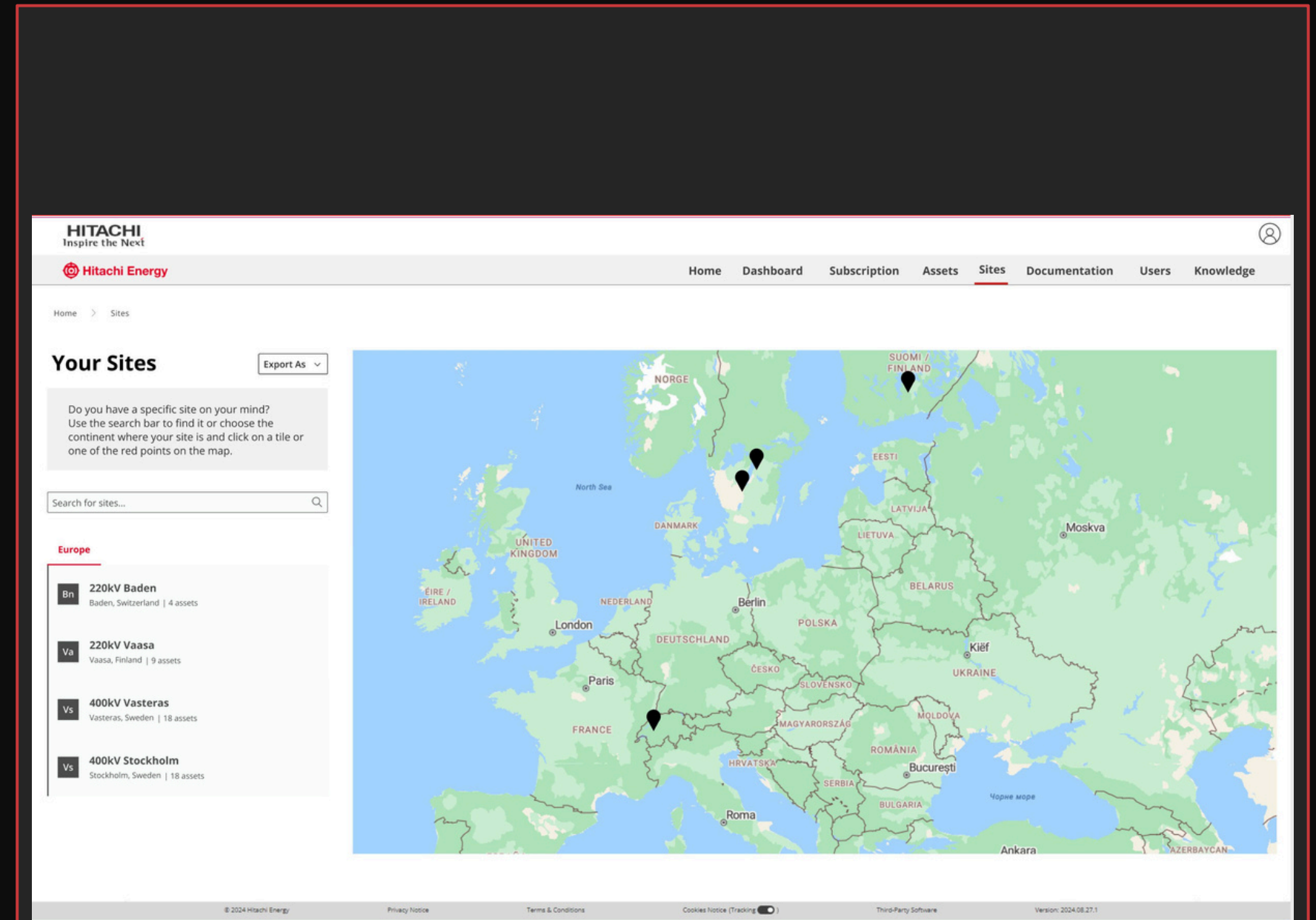
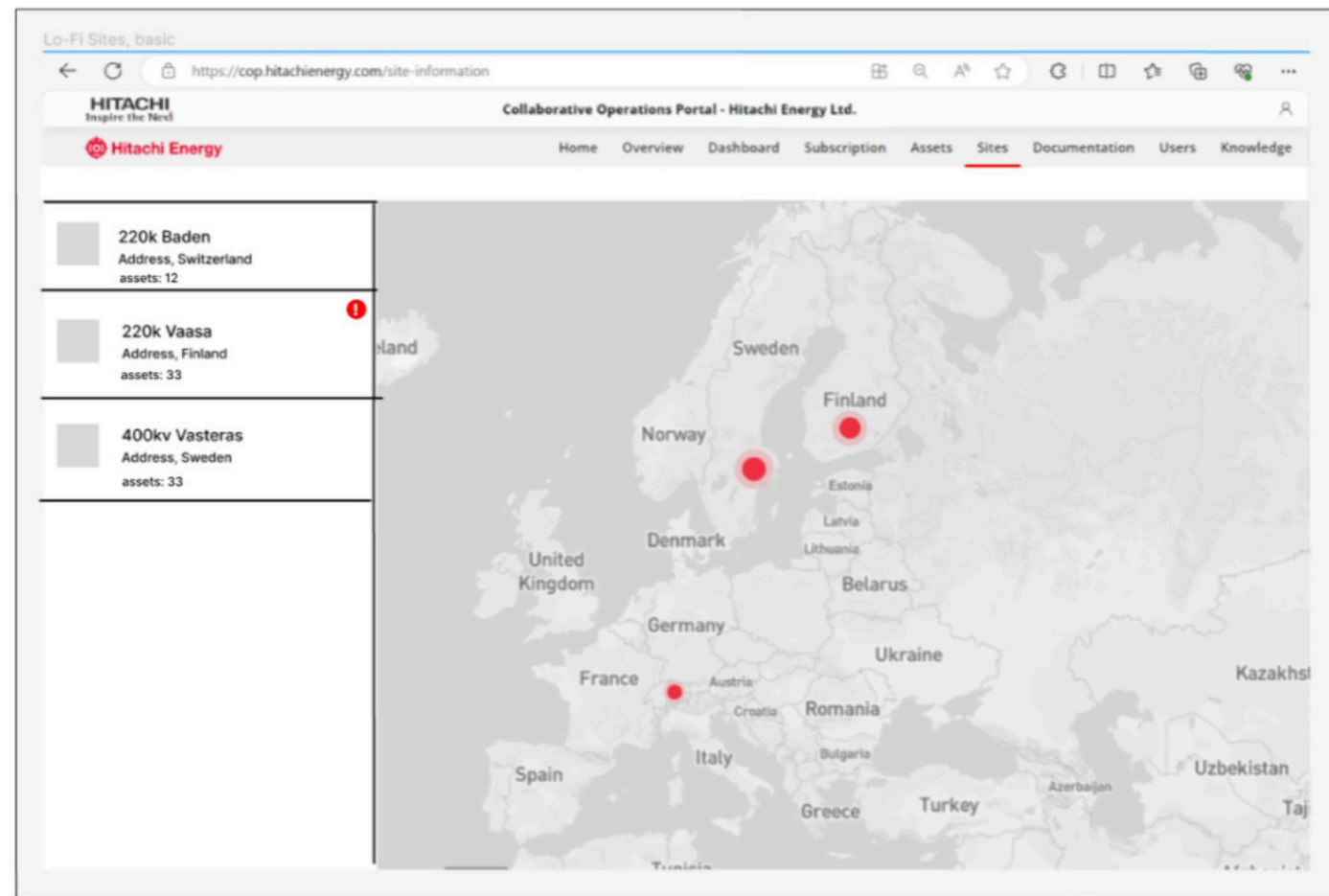
Sites developed into the cleaner of the two workstreams. The hierarchy mapped naturally onto geography: overview map → site list → site detail → connected assets. I designed the structure before deciding how it would look — wireframes stayed intentionally rough until the logic was agreed.

LO-FI WIREFRAME

AFTER — FINAL SITES UI

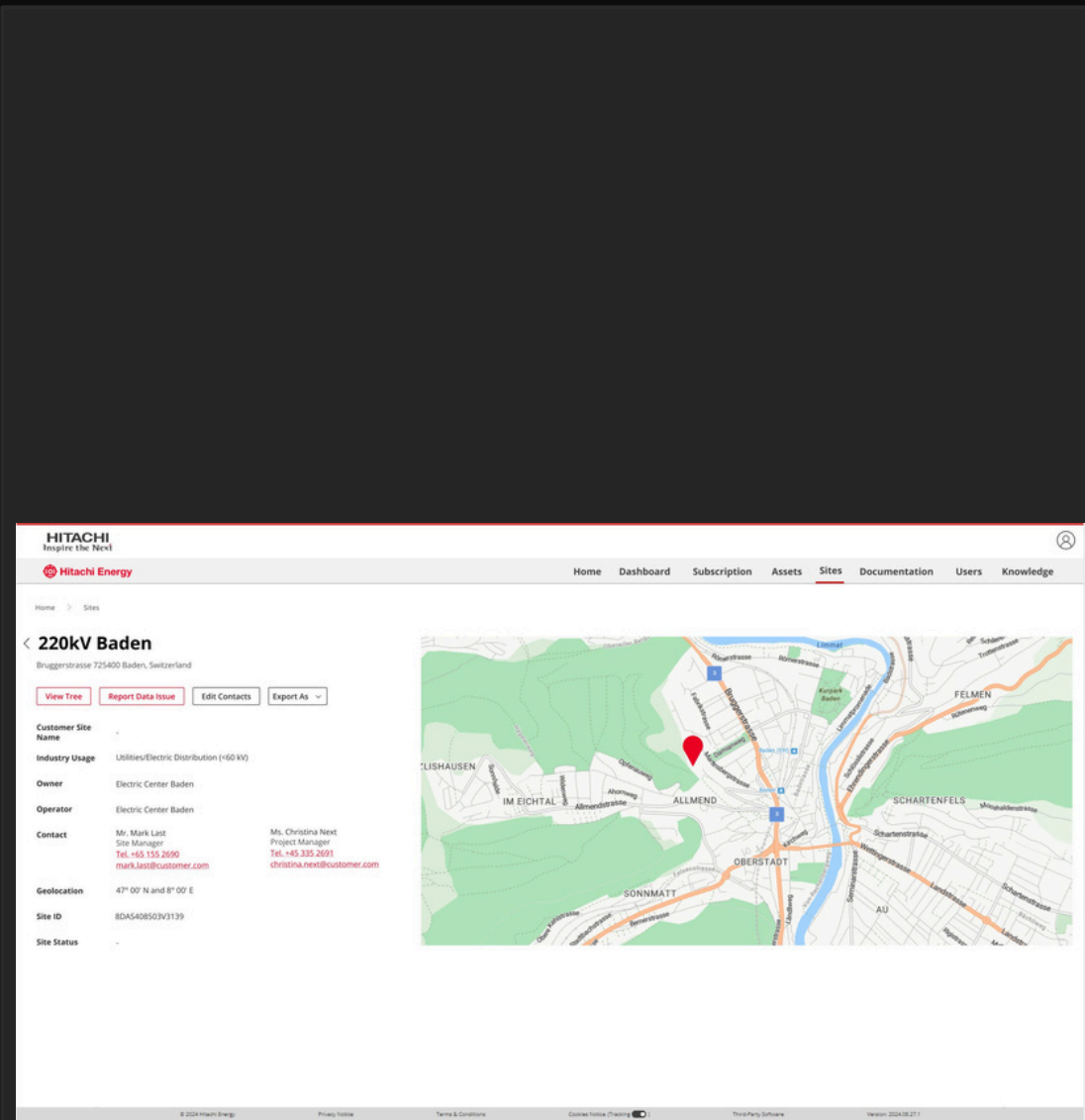
First Wireframes

Sites - part 1

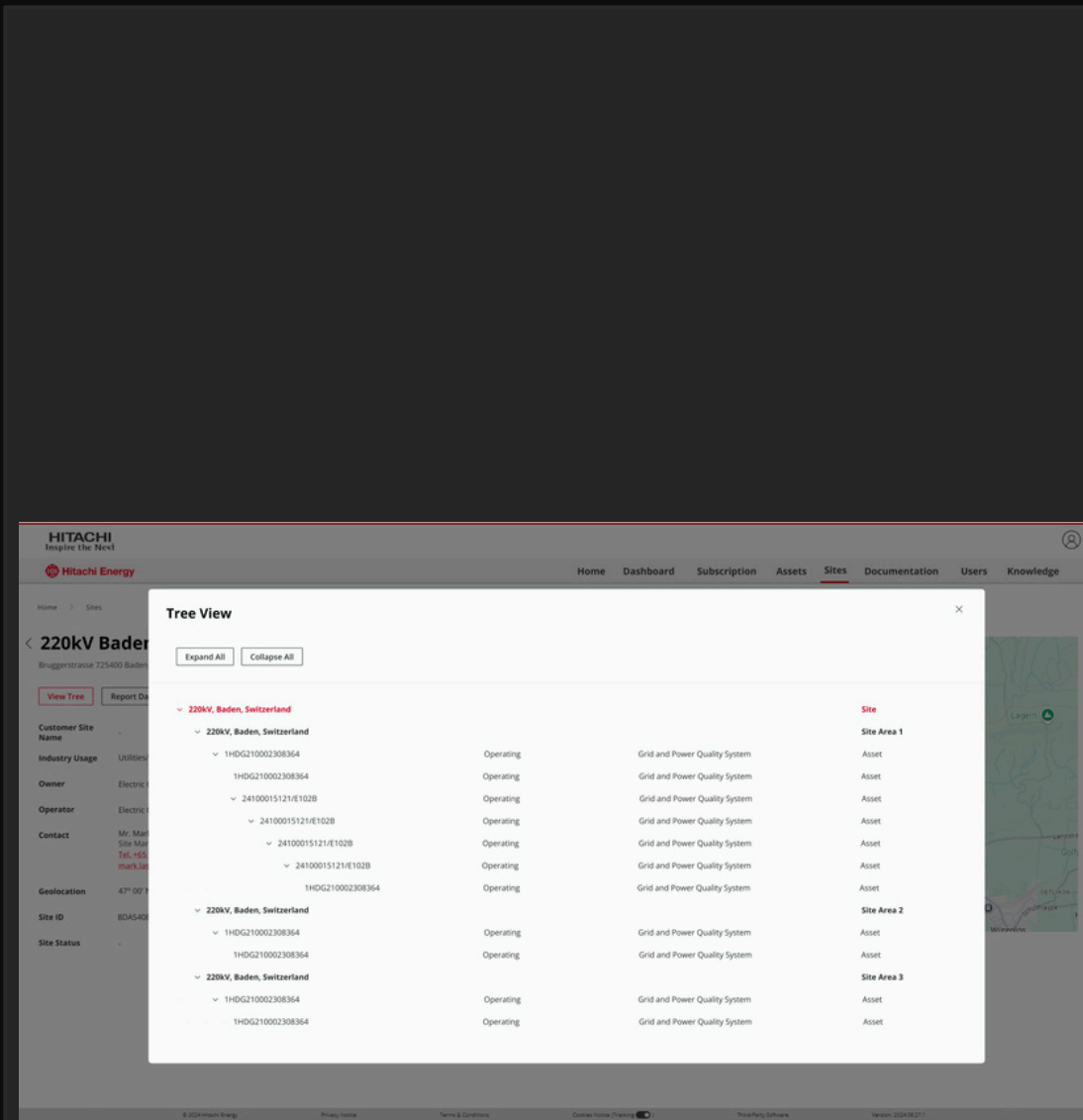


Three screens that show the depth behind the map.

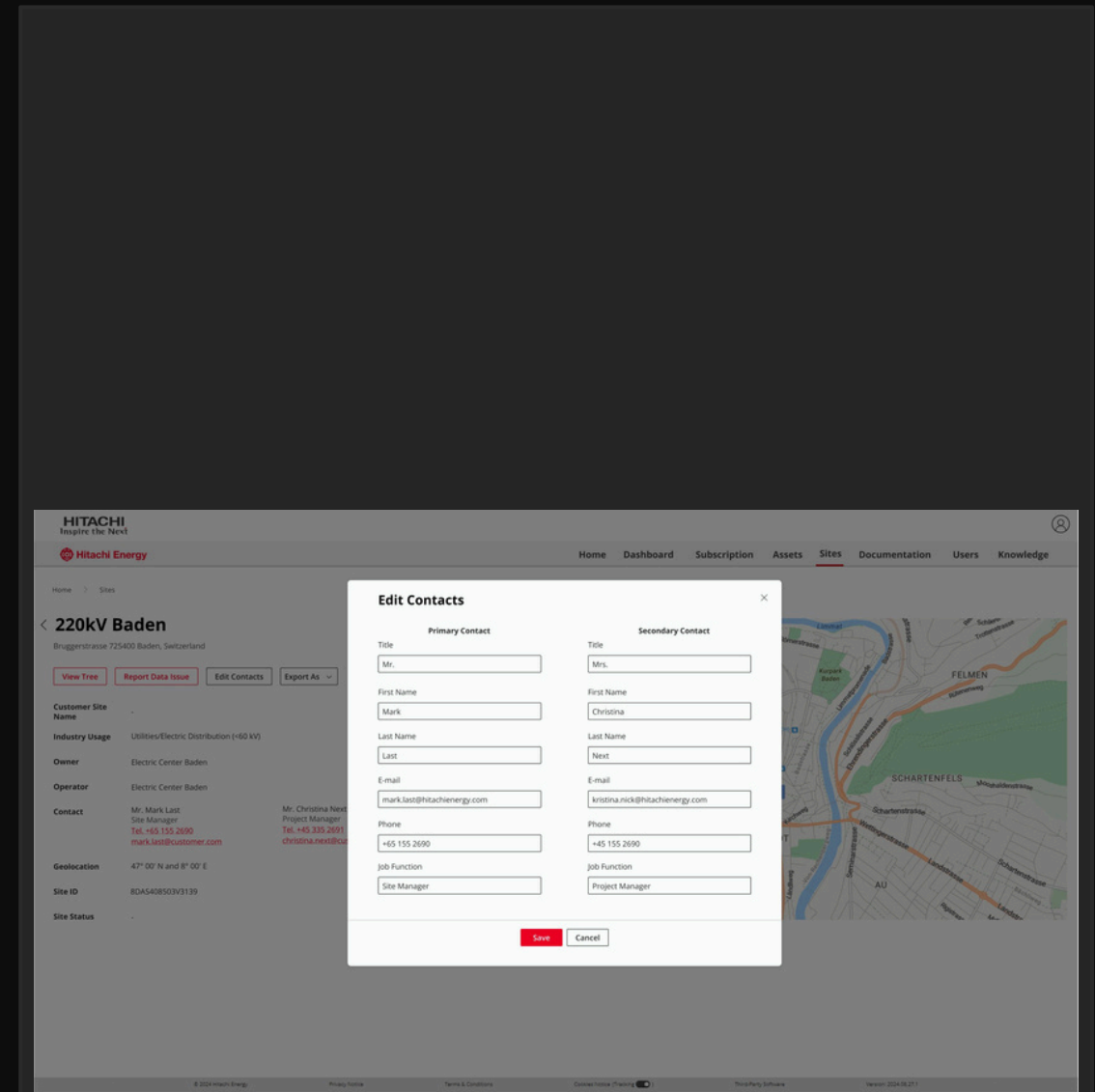
Site detail page



Tree view modal



Edit contacts modal



Address, contacts, geolocation, status, and key actions. Button order and prominence were deliberate decisions — not defaults.

Full asset hierarchy for a site: Site → Site Areas → Assets. Gives internal users a structural overview without leaving the page.

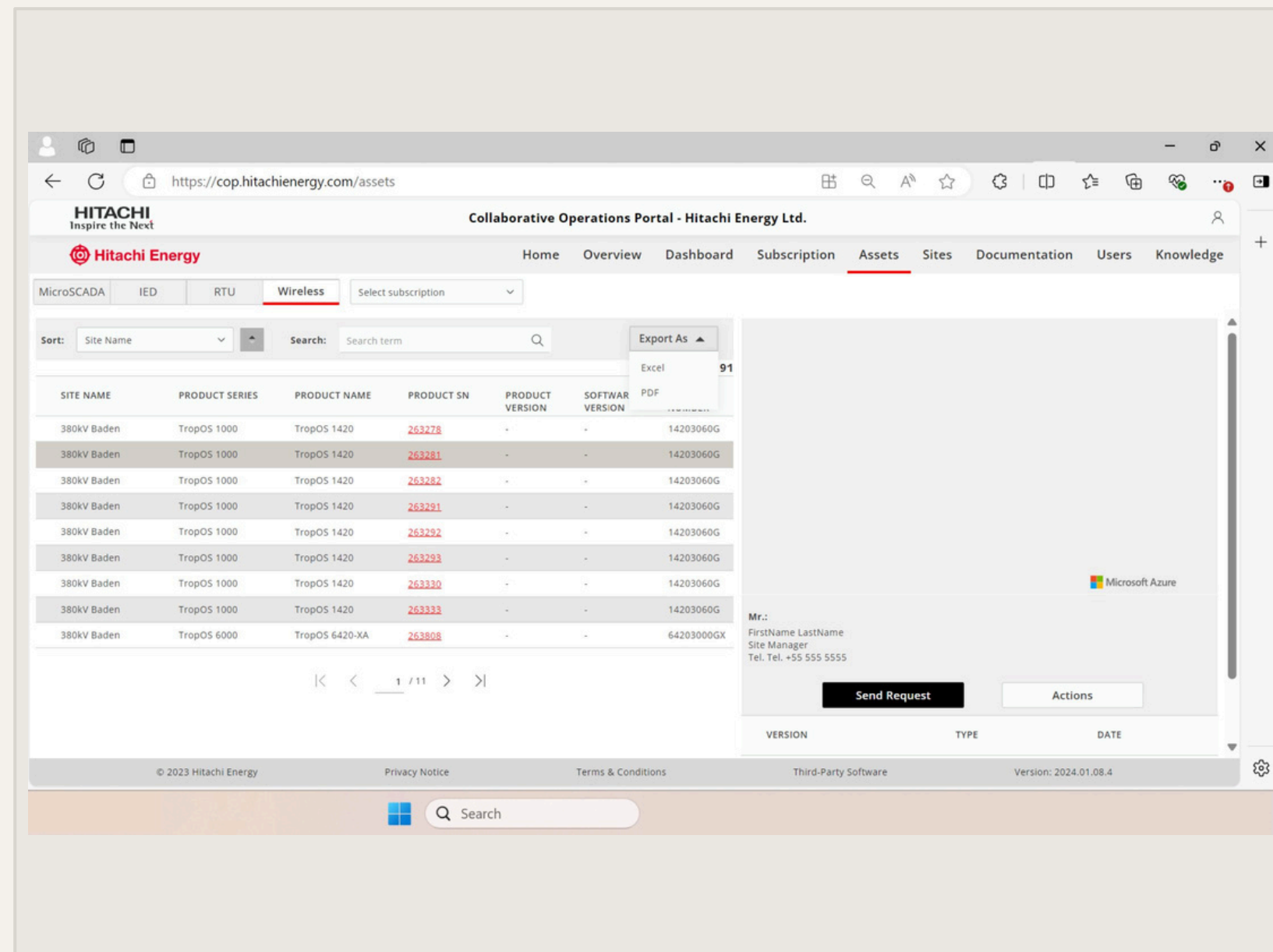
Primary and secondary contact management. MVP scope — designed with room to grow as the portal matures post-launch.

Redesigning the scope, not just the screens.

The old Assets page was a flat tab-filtered list built only for Grid Automation — a single BU, a split-panel layout, no catalog structure. My redesign introduced a Product Catalog entry point that lets users navigate by business unit and product category before reaching any list. That structural decision prevented a single page from collapsing under multi-BU weight.

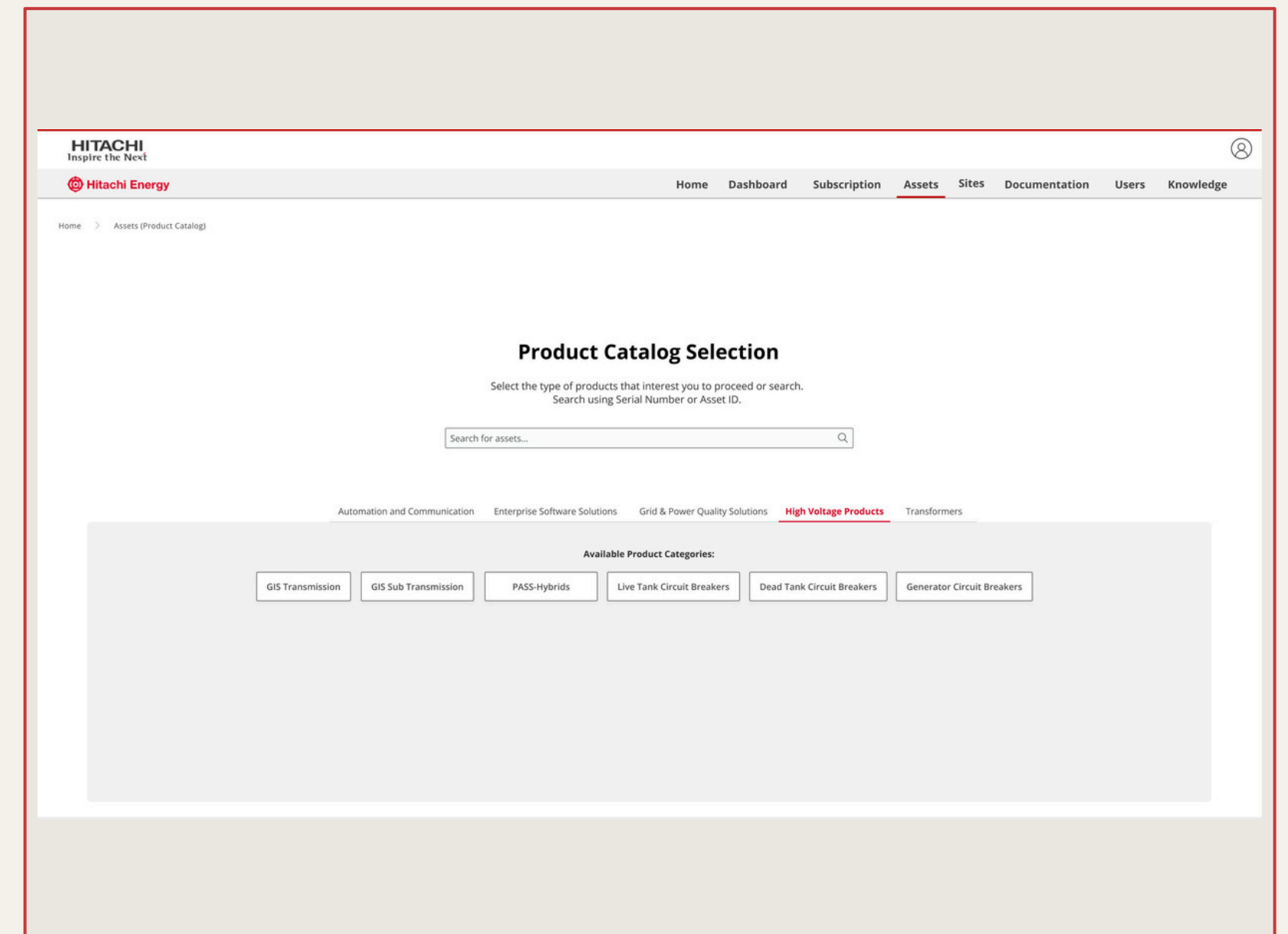
BEFORE

Legacy Assets — Grid Automation



AFTER

Redesigned Assets — Product Catalog first



From catalog to asset list to individual asset detail.

Asset list — HVDC Transformers

The screenshot shows the 'HVDC Transformers' asset list page. It features a search bar, an 'Export As' dropdown, and a table with columns for Serial Number, Status, Site Name, Asset Name, Asset ID, Customer Tag, and Power Base/Top (MVA). Each row includes a 'View' button. The table is paginated, showing 18 assets in total.

Serial Number	Status	Site Name	Asset Name	Asset ID	Customer Tag	Power Base/Top (MVA)	
8720855	Operating	220kV Baden	H221-QT1	88K484432L1771	-	100/120	View
7720854	Limited	220kV Baden	H331-QT1	9AAK484432L1771	-	-/-	View
7730849	Limited	220kV Baden	H411-QT1	6AAK484432L1771	-	-/-	View
6223.1737	Limited	220kV Baden	G111-QT2	5AAK484432L1771	L2	-/-	View
45223.1739	Operating	220kV Baden	G122-QT3	5784484432L1771	-	-/120	View
4223.1740	Operating	220kV Baden	G344-QT1	48748GGG4948	L2	100/-	View
9923.1735	Operating	220kV Baden	G355-QT3	5AAK484432L1771	L2	-/120	View
1123.1734	Limited	220kV Baden	G4984-QT5	48748GGG4949	-	-/-	View
3223.1733	Operating	220kV Baden	VSC-V	48748GGG4948	-	-/-	View
2223.1732	Operating	220kV Baden	VSQ-V	5AAK484432L1771	-	100/120	View
81111.1731	Operating	220kV Baden	VSCF	367347BAACOP99	L2	100/120	View
83938.1731	Operating	220kV Baden	VSVVB	22FHAOU35784909	L2	100/-	View

After selecting a product category, users see a paginated list with serial number, status, site name, asset ID, and

Asset detail — Product Specification

The screenshot shows the 'Product Specification' page for asset 8720855. It includes a 'Report Data Issue' button, a map of the site location, and a table of technical details.

Product Specification
Serial Number: 8720855

[Report Data Issue](#)

Site Name: 220kV Baden
Status: Operating
Asset Name: H111-QT1
Asset ID: 9AAK560165U8897s
Customer Tag: -
Power Base/Top (MVA): 100/120

Details

High Voltage (kV)	Low Voltage (kV)	Age	Manufacturer	Delivery Date
-	-	2	Hitachi Energy	02.02.2022

Individual asset view: lifecycle status, ordering code, available spares, and field upgrade history. Deliberately

My design decisions — what wasn't in the business requirements

Controlled use of red as an emphasis system

Red was a company brand colour — but that made the choice more nuanced, not simpler. I used red selectively for actions that deserved stronger prominence. The goal was hierarchy and brand alignment without visual aggression or ambiguity. Not all buttons are red. That restraint is the signal.

Intentional minimalism on MVP pages

Site Details and Product Specification may look relatively open at MVP. That was deliberate. I knew these pages would gain functionality later. Keeping layouts open meant future content could arrive without forcing a full redesign. The value was partly in what the structure made possible — not just what was visible at launch.

Column order and button sequence matter in enterprise

In tables and detail pages, I made explicit decisions about column order, button placement, information sequencing, and what feels primary vs secondary. In enterprise interfaces, these decisions determine what is discoverable and what gets lost. They rarely appear in a requirements document but shape daily use.

The Site Details agreements table — proof of concept

After the MVP, I added a related agreements table to Site Details: internal users needed to see which sites were linked to which agreements. Because the original layout had been kept deliberately open, the new block fit naturally without restructuring. This is direct evidence that the early restraint paid off.

What made this hard — and what it produced.

WHAT MADE IT HARD

No defined user model at the start of the project.

6+ BUs with competing views, all involved in design decisions at once.

Difficulty reaching real end-users early — most input came through stakeholder interpretation.

Assets kept accumulating adjacent scope, creating continuous overload risk.

Parallel rebranding pressure increased complexity throughout.

WHAT IT PRODUCED

Ambiguity → structure

Stakeholders could respond to concrete proposals instead of abstract requests. Weekly reviews became more focused.

Overload risk named early

Card sorting exposed the Assets problem before it was built. Naming the risk is itself a design output.

Scalable page design

The Sites experience moved from a limited legacy view to a clear, expandable hierarchy.

Proven future-proofing

The agreements table addition proved the original restraint worked — no redesign needed for new content.

Org. learning

The team learned why broad discovery and broad decision-making are not the same thing.

The most senior part of any case study.

These are not project-management complaints. They are enterprise UX insights — patterns that will shape how I approach the next complex, multi-stakeholder product initiative.

01

Broad discovery ≠ broad decision-making

Using many BUs for discovery is valuable. Using all of them equally in detailed weekly design decisions creates friction, delays, and instability. The right model: wide input early, narrow decision group for detailed execution. The process owner later acknowledged this explicitly.

02

Sites and Assets should have been split earlier

They shared an umbrella — Installed Base — but had fundamentally different complexity, pace, and scope risk. Running them together in one workstream added overhead and contributed to delay. Separate milestones and validation rhythms would have served both better.

03

Push for direct user access sooner

Stakeholder interpretation is not user validation. Even limited customer contact early would have grounded the direction more reliably. When real customer feedback eventually arrived, it was more useful than weeks of BU debate.

04

Intentional restraint is a design decision — not an omission

Pages that look minimal at MVP are not unfinished. They are designed for growth. The agreements table story proves it: the earlier open structure absorbed new functionality naturally, without requiring a rebuild. Future-proofing needs to be planned at the start, not retrofitted later.