

FEDERICO BACCETTI

Senior UX/UI & Service Designer

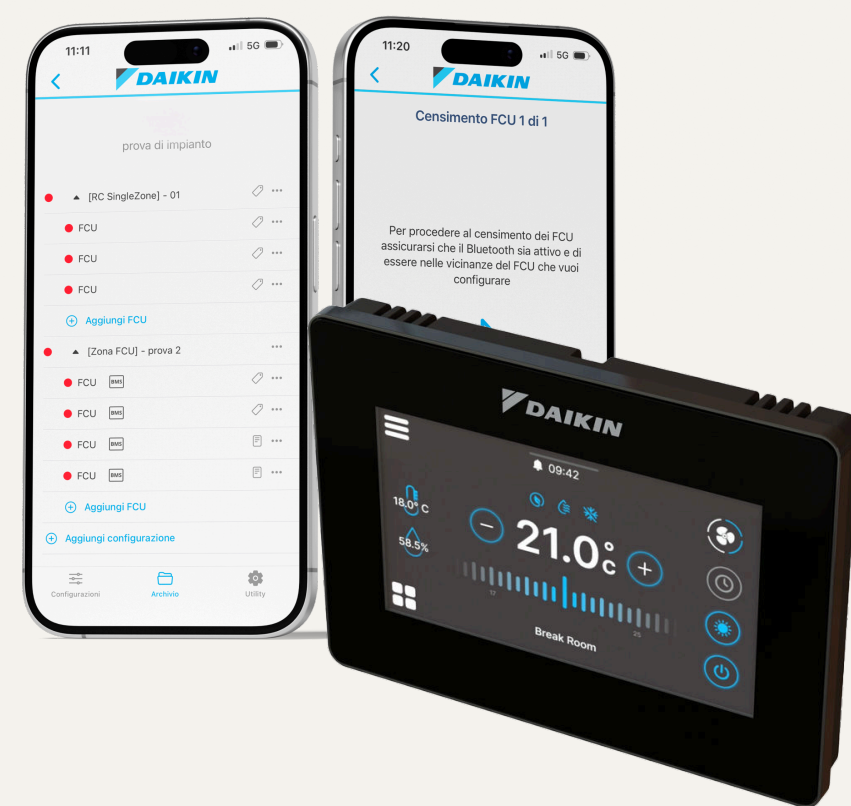
I design interfaces, services, and experiences capable of connecting people, processes, and business goals. I have experience in complex industrial contexts, where design becomes a tool to make solutions clearer, more accessible, and scalable.

I'm a designer constantly exploring creative solutions to improve people's experiences.

I design solutions with the objective of concretely improving the experience of the people who will use them. I specialize in complex industrial solutions where technological innovation and business growth must find a concrete balance.

- UI for HMI and complex industrial systems
- B2B SaaS & enterprise platforms
- Design system & UI architecture
- AI-assisted prototyping

Case studies



01

DAIKIN Shinka Ecosystem

HMI and Mobile fancoil control ecosystem

Design of software with an HMI interface for DAIKIN's new fancoil control solution, unifying technical workflows between the HMI controller and mobile application.

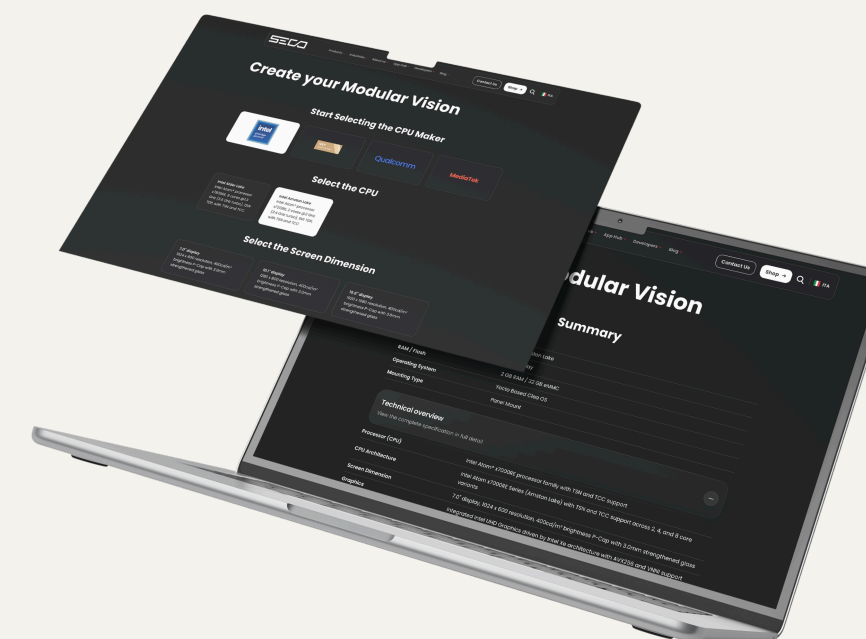


02

Clea Software Suite

Industrial software framework for IoT and Edge AI infrastructure

Design of a scalable digital ecosystem to simplify the management of fleets, data, and Edge AI services within an integrated industrial platform.

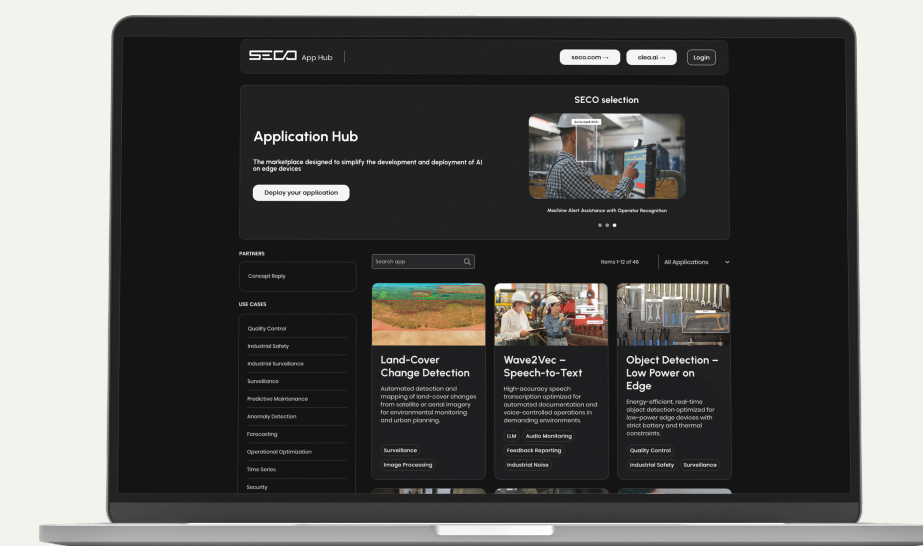


03

SECO Online Configurator

Online configurator for hardware products

Design and development via AI coding of an online configurator for catalog families of modular hardware products.



04

SECO Developer Experience

B2B documentation platform

Conception and design of a digital hub to centralize documentation, guides, and technical resources for the entire SECO hardware and software ecosystem.

DAIKIN

ECOSYSTEM

Project Overview

CONTEXT

Daikin Applied Europe needed to design a new solution for fancoil control in large multi-zone systems. The product featured a 4.3" **touch interface** managed by a control PCB, capable of supporting advanced logic for optimizing and managing the fancoil unit network. A **mobile Web App** dedicated to remote pre-configuration was also planned.

APPROACH

I was the UX lead primarily for the **Room Controller** side, working on technical requirements, flows, information architecture, and interface hierarchies. HMI and App were designed with consistent but context-adapted logic, through reviews with specialized technicians and careful attention to **embedded device** implementation constraints.

CHALLENGE

Designing from scratch **two complementary interfaces** for a complex technical system. Room Controller and App had to make fancoil configuration, control, and management simple, while maintaining continuity between the embedded experience and remote pre-configuration, **all within hardware constraints**.

IMPACT

Ease of configuration, improved workflow for installer technicians, **reduced on-site time** thanks to saved pre-configurations, and the delivery of a **feature-rich**, easy-to-use Room Controller user interface. By creating a common logic between local control and remote configuration, the experience better supports the work of installers.

Architecture & Concept

Two interfaces, one system logic

Room Controller and App were designed with mirrored architectures, to ensure continuity between local control and remote configuration.

Consistent actions, different modes

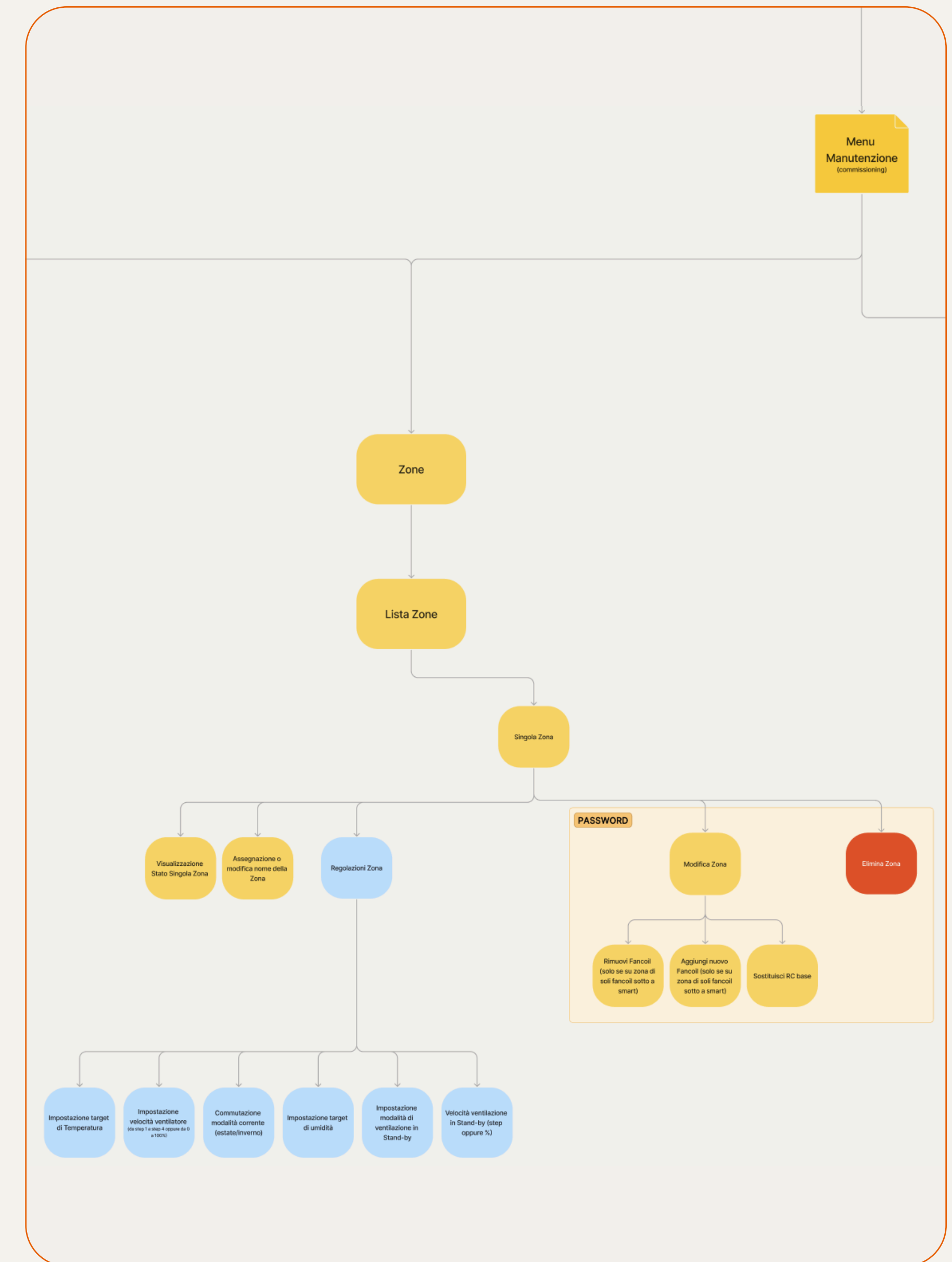
The same core functions were made available on both touchpoints, adapting flows, interactions, and priorities to the usage context.

Permission management

The architecture distinguishes between installer and end user, separating technical configurations, advanced controls, and everyday functions.

Hardware-constrained design

Navigation and interaction logic were defined taking into account the performance and constraints of the Room Controller, prioritizing simple paths, clear hierarchies, and reduced operational load.



Daikin Shinka Controller

The operational core of the system, Shinka Touch was designed as a **local interface** for **fancoil control and management**. The design had to balance hardware constraints, limited memory, and the need for smooth interaction. The **LVGL framework** was leveraged to build a clear and coherent interface, maintaining a pleasant look and feel and an intuitive UX without burdening the device.

	Role	UX/UI lead, analysis, interviews, flow definition, UX/UI design of final mockups.
	Deliverable	Information architecture, wireframes, final mockups, and UI guidelines for LVGL implementation.
	Goal	To create a lightweight, clear, and fluid local interface for fancoil control, respecting the hardware limits of the device, a 4.3" HMI.



OUTCOME

A lightweight, clear, and consistent UI to make local fancoil control immediate and accessible, within the hardware constraints of the Room Controller.



Daikin Shinka Manager

A **mobile application** designed to support **installers and technicians** in configuring the system before on-site intervention on the physical device. The app allows structures, zones, and parameters to be prepared more flexibly, **reducing dependence** on the Room Controller during operational activities.



Role

UX/UI design of the app, from flow definition to the creation of final mockups.



Deliverable

Experience architecture, user flows, wireframes, high-fidelity UI, and development specifications.



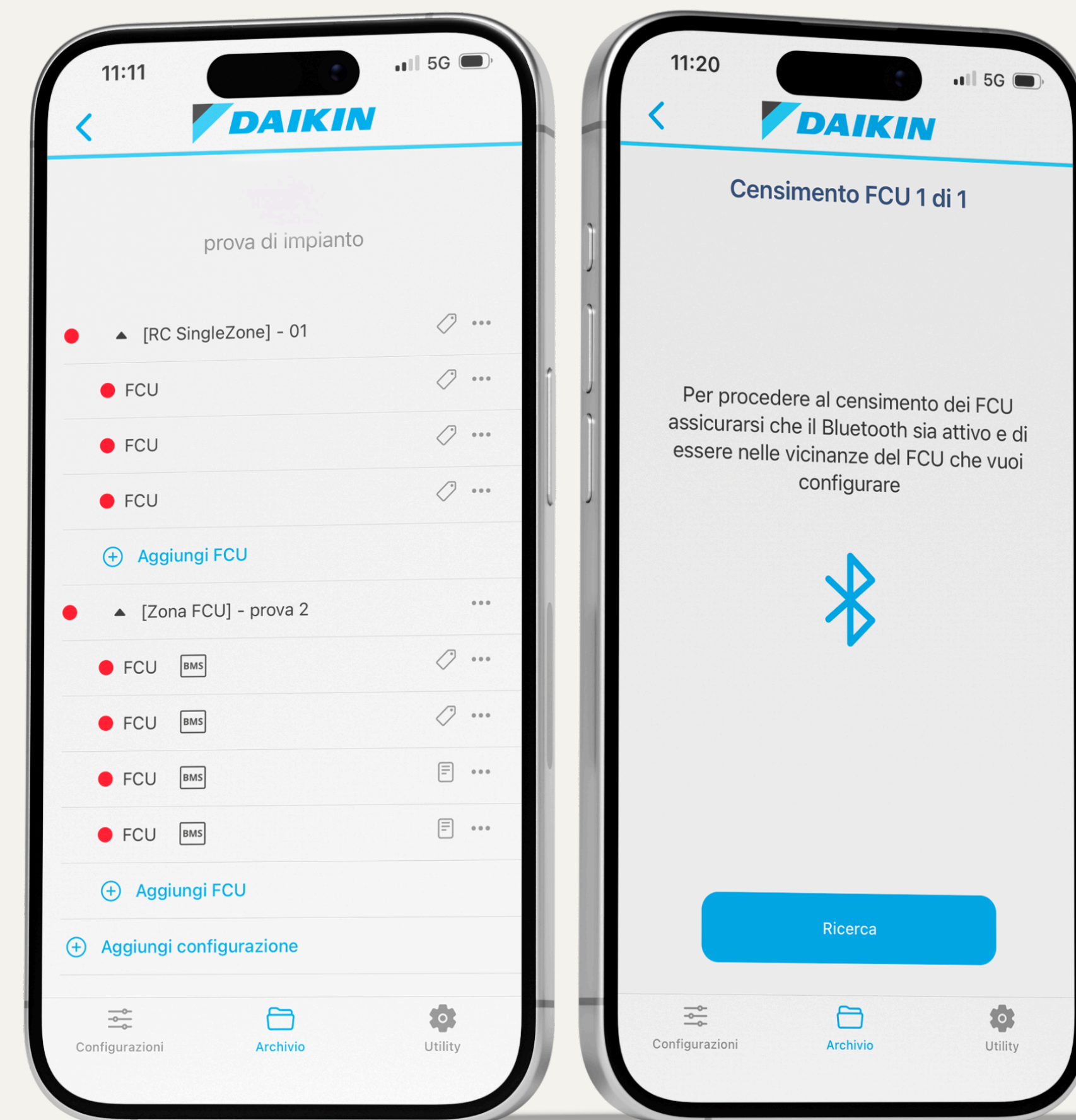
Goal

To create a mobile interface consistent with the Room Controller, capable of simplifying configuration, management, and data synchronization for technicians.



OUTCOME

A mobile app designed to extend the Shinka ecosystem beyond the physical device, enabling a remote workflow and simplifying on-site intervention and local control.



PROJECT VALUE

Value Generated

The project shaped two complementary touchpoints to manage local control and remote configuration, making clearer the technical flows that previously would have remained tied to the physical device alone.

- HMI UI designed around RC constraints
- App for on-site configuration
- Continuity between local and remote control

Skills Applied

The work required translating complex technical logic into usable interfaces, balancing functional requirements, hardware constraints, and the operational needs of installers and end users.

- Turning technical requirements into UX flows
- Flow architecture between HMI and App
- UX/UI for embedded and mobile devices

Contribution

As the UX/UI lead, I was responsible for defining the UX architecture between Room Controller and App, separating roles, permissions, and usage contexts to make local control and remote configuration consistent.

- UX architecture between HMI and App
- Installer / end-user role management
- UI adapted to hardware constraints

CLEA SOFTWARE SUITE

Project Overview

CONTEXT

Clea is a **software suite** built to offer digital services with **added value** on hardware devices. The platform brings together several modules dedicated to connectivity, device management, data, AI applications, and monitoring, with the goal of creating a **unified ecosystem** to manage complex industrial scenarios and enable new data-driven service models.

APPROACH

I worked on defining **shared UX/UI logic** for the suite and was **owner** of specific areas, including Clea Console, which I designed end-to-end. The work covered navigation, dashboard, components, operational flows, role-differentiated views, and consistency with the Clea design system.

CHALLENGE

The complexity was not only visual, but **structural**: the UX had to hold together **diverse products**, diverse roles, and very different levels of technical depth. Each module had to work independently, without feeling disconnected from the rest of the suite.

IMPACT

The result is a **digital ecosystem** coherent and modular, capable of supporting the management of devices, data, and Edge AI applications through a **clear and consistent experience**. The suite thus becomes a scalable foundation for building, monitoring, and evolving industrial IoT solutions.

Clea Design System

A shared system for the Clea ecosystem

Custom design system to support a modular software suite, maintaining visual and functional coherence across products, sections, and diverse platform flows.

Reusable components and patterns

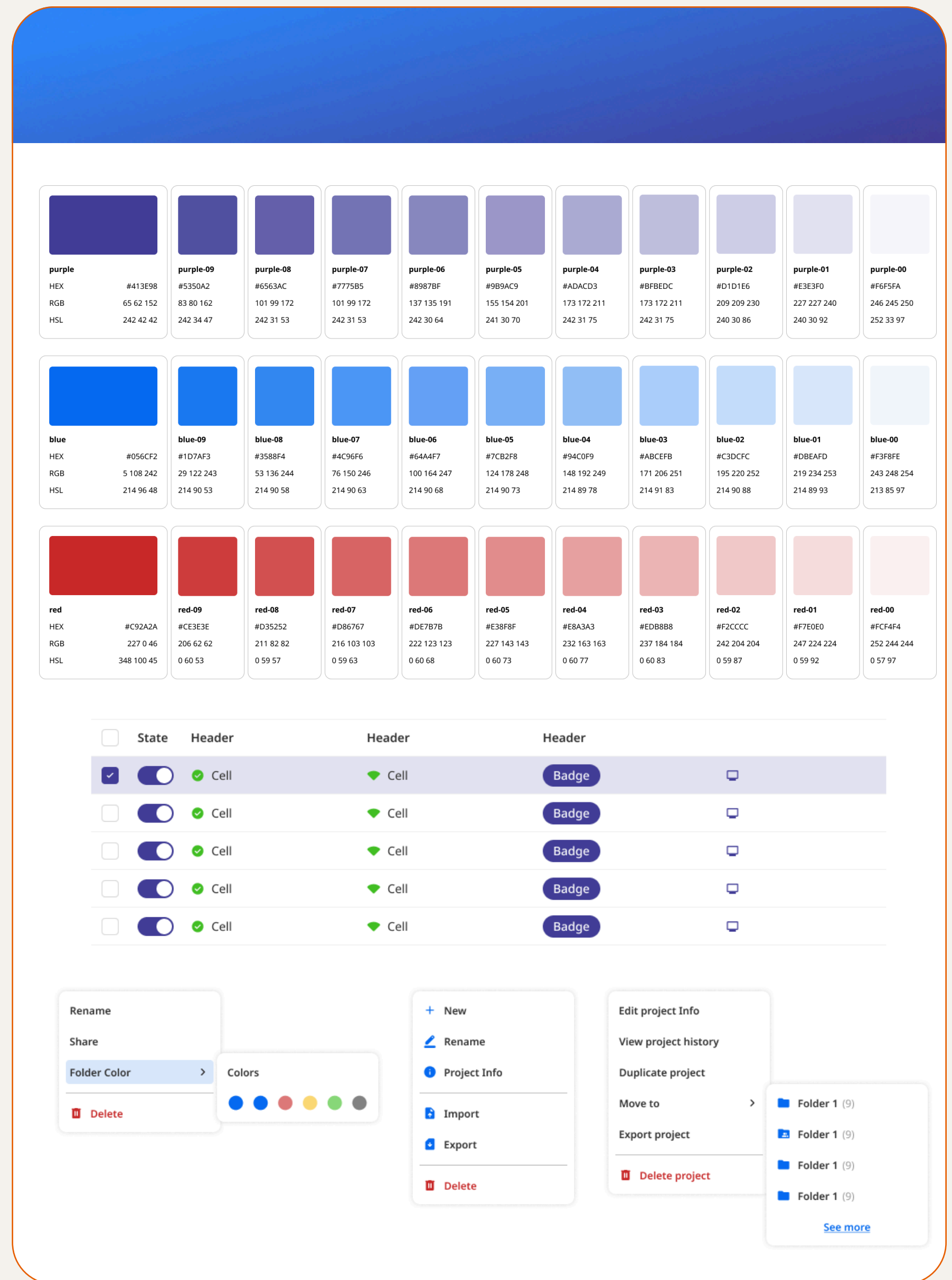
The system defines components, states, variables, and design patterns built to adapt to **different usage contexts**, ensuring continuity across the various areas of the Clea ecosystem.

Accessibility and consistency

Colors, typography, states, components, and guidelines were defined taking into account **WCAG principles**, with attention to readability, contrast, usability, and interface scalability.

Design to development

The design system structure was conceived to make the handoff from mockups to code clearer, creating **a shared foundation** between design and development for reusable, maintainable, and evolving interfaces.



The Clea Ecosystem

Designing an ecosystem

Working on Clea required a **systemic vision** capable of holding together products, modules, and diverse touchpoints. The design goal was to build a coherent, recognizable, and scalable experience within a **complex software suite**.

Making complexity readable

Technical flows, operational data, configurations, and advanced functions were translated into **clearer and more usable paths**. The design focused on information hierarchies, navigation, and patterns capable of reducing cognitive load for users with different needs.

A shared language across products

Dashboards, modules, operational tools, and dedicated views were designed with shared logic, **ensuring continuity** across the ecosystem's areas. The design system consolidated components, states, and behaviors into a common foundation.

System, service, interface

The goal was to connect processes, roles, data, and interactions into an **end-to-end experience**. The result is a design structure capable of supporting the suite's evolution and making the transition between business needs, technical requirements, and final interfaces smoother.



Clea Console

One example of the many parts that make up the Clea ecosystem is **Clea Console**. Designed as a **governance platform** for the SECO software ecosystem, it centralizes operational control, client management, contracts, instances, usage metrics, SLA, and billing.

The work required defining the logic, flows, and UX for a product intended **for both internal teams and clients**, with differentiated views based on permissions and roles.



Role

Project ownership, UX/UI design of the entire platform, from flow and information architecture definition to the creation of final mockups.



Deliverable

User flows, user stories, experience architecture, management modules, client views, UI components, and high-fidelity mockups.



Goal

To create a single governance platform for managing data, contracts, billing, SLA, and instances.



OUTCOME

A modular platform that makes SECO software service management clearer, offering internal teams operational control tools and clients a more transparent view of instances, usage, SLA, and service information.



PROJECT VALUE

Value Generated

The project defined a shared design foundation for a modular software suite, capable of holding together products, services, user roles, and diverse touchpoints within the same ecosystem.

- A shared logic for the entire ecosystem
- Continuity across tools and operations
- Foundation for new module integration

Skills Applied

The work required structuring an ecosystem made up of diverse modules, defining shared logic for navigation, dashboards, data management, and UI components.

- UX/UI for modular enterprise platforms
- Information architecture and service logic
- Design system for modular products

Contribution

I contributed to structuring a shared UX/UI foundation for the entire suite, defining navigation logic, components, and cross-module patterns, with ownership roles over specific project areas such as Clea Console.

- Common navigation for modular products
- Reusable UI components across areas
- Scalable structure for new modules

ONLINE HARDWARE CONFIGURATOR

Project Overview

CONTEXT

SECO needed to create an **online configurator** to support the selection of modular hardware products. The goal was to **guide the user** through the composition of a configuration based on technical parameters such as vendor, processor, memory, storage, operating system, and mounting type.

APPROACH

Holding **full ownership** of the project, I managed UX, UI, configuration logic, and frontend prototype. I defined the step-by-step flow, the choice hierarchy, the summary page, and product cards. The mockups were then transformed into a working prototype via AI-assisted development, using Codex as operational support for **frontend development**.

CHALLENGE

The risk was turning a complex technical catalog into a slow, dispersive experience understandable **only by expert users**. The configurator had to guide selection step by step, prevent inconsistent choices, keep the final configuration readable, and highlight standard alternatives already available.

IMPACT

The result is an interactive configurator that makes **more accessible** the selection of modular hardware. The user can build a configuration, view a technical summary, compare similar products, and reach **more easily** the commercial request, reducing the typical dispersion of a very wide catalog.

Design System

Integrated into the seco.com ecosystem

The configurator was designed to be hosted within the official SECO website, maintaining **visual continuity**, tone, and behavior with the existing digital experience.

Consistency with the SECO design system

Layouts, components, states, typography, and interactions were defined following the **guidelines** of the site's design system, built by the team to ensure consistency across corporate pages, product areas, and digital tools.

Common patterns

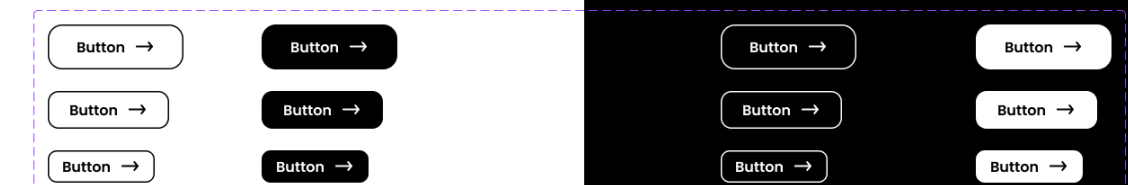
The design favored the reuse of patterns and components already present in the SECO ecosystem, simplifying the transition **from mockups to development** and keeping the configurator scalable, consistent, and easily integrable into the site.

Button

Primary



Secondary



Tertiary



Icon button



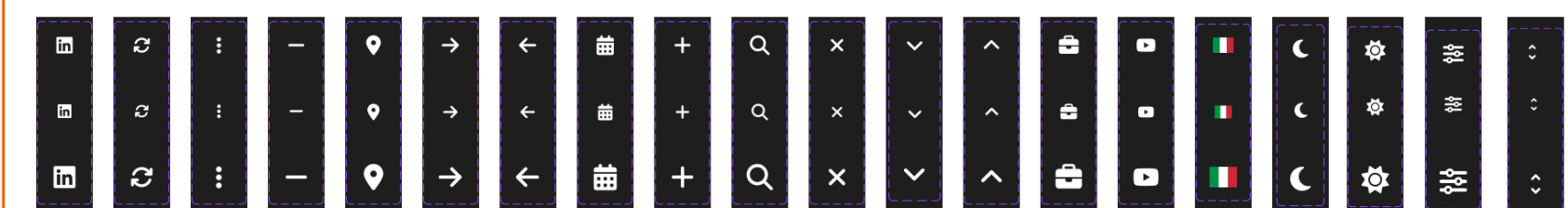
Link



Application Icons



UI Icons



Frontend Design and Development

Mockups as a design foundation

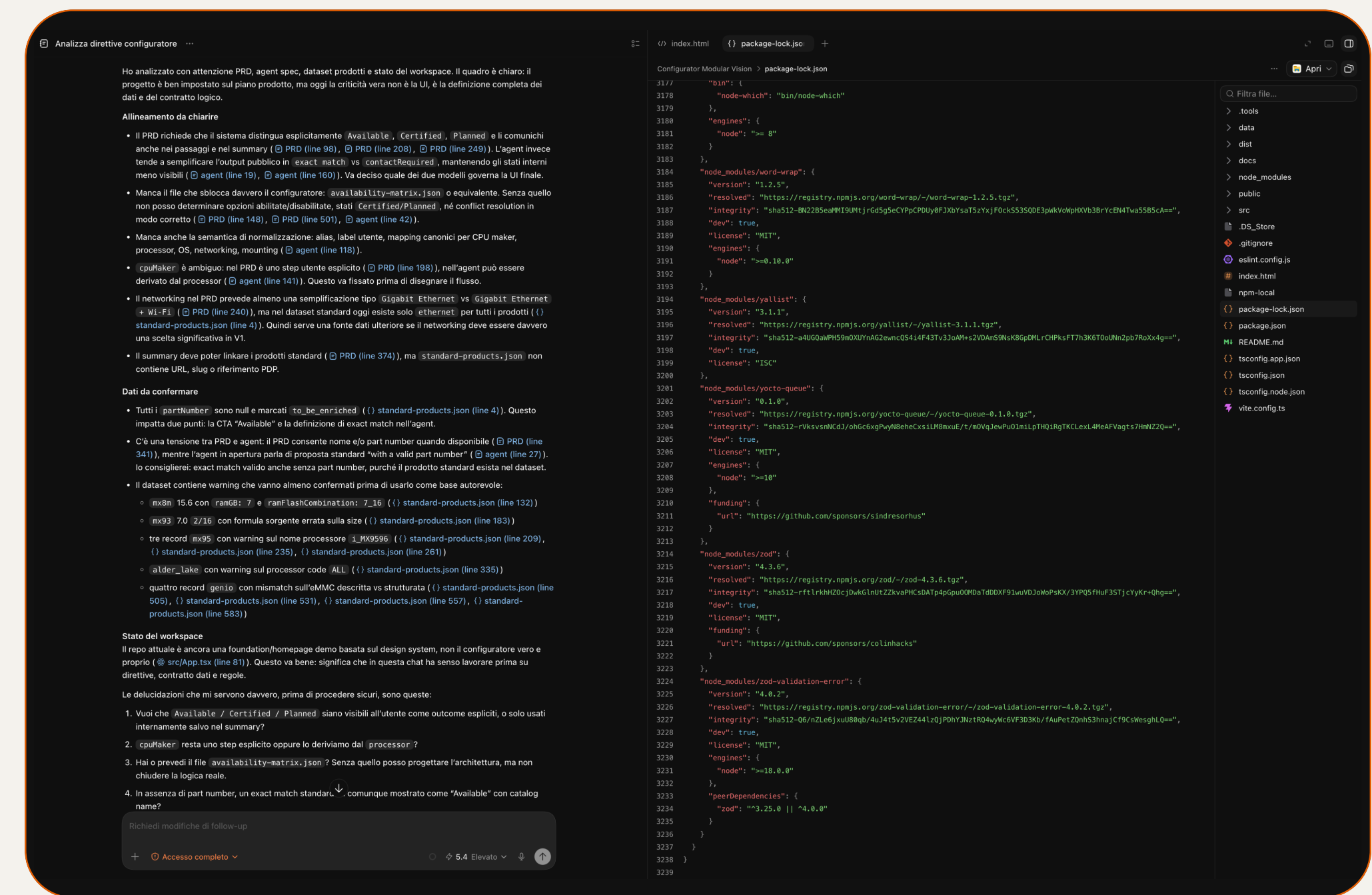
The process started from **mockups** at high fidelity, used as reference to define the layout, hierarchies, states, components, and behavior of the configurator.

Frontend Development with AI-Assisted Development

The configurator was developed through **AI Prompting** with the support of Codex, turning design decisions into a **working prototype** with step-by-step flows, selection logic, technical summary, and similar product suggestions.

Consistency between logic and interface

Design and development were carried out in an **integrated and continuous** manner, allowing rapid validation of interactions, configuration constraints, and interface behavior, maintaining **continuity** between the designed experience and the implemented product.



Online Configurator

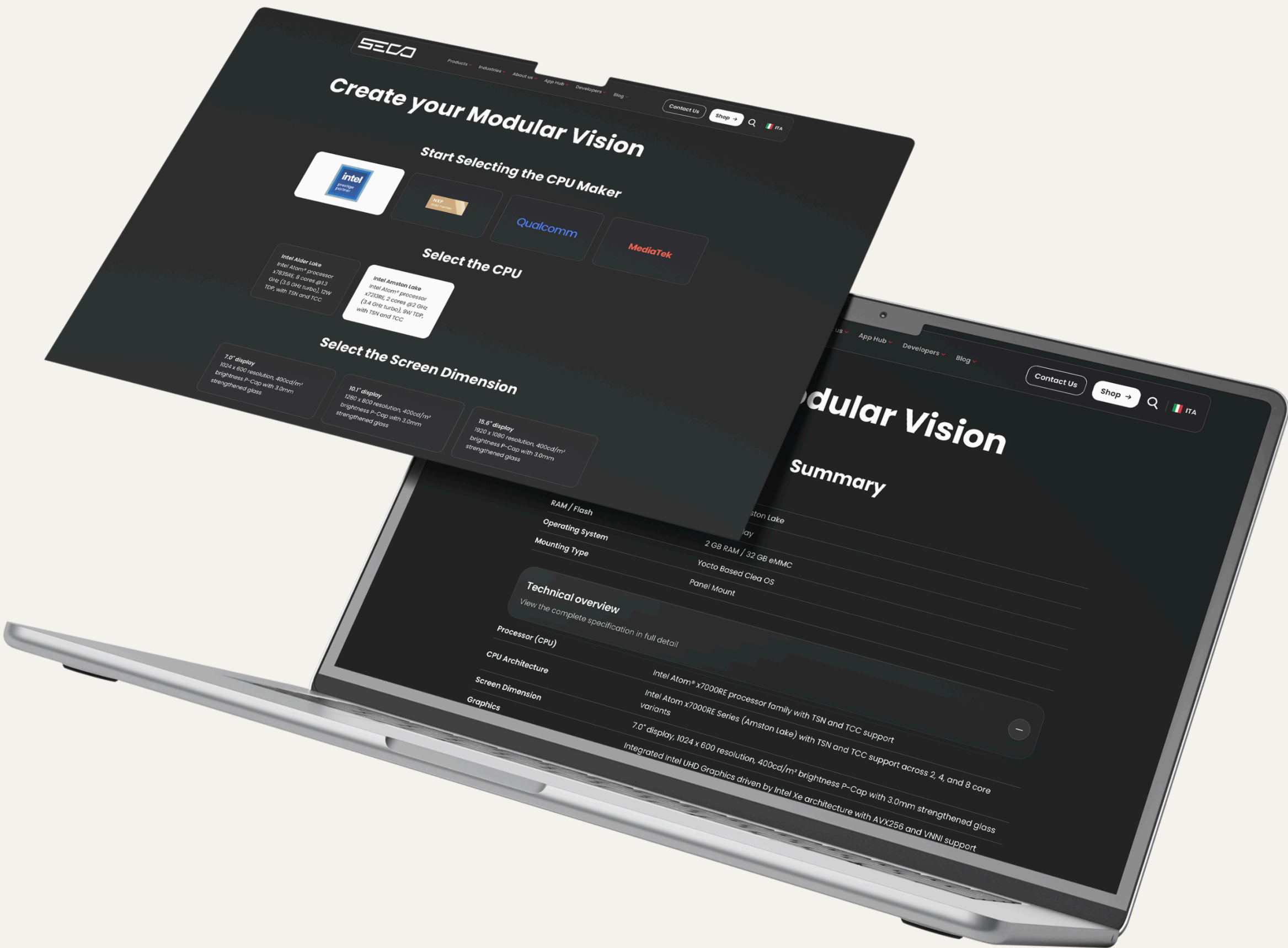
Web configurator designed to guide the user in composing a modular hardware product, starting from the selection of key components. Built from experience definition to interface implementation, through **AI-assisted frontend development** with the support of Codex.

	Role	Ownership, UX/UI design, flow definition, and development of the working prototype.
	Deliverable	User flows, configuration logic, step-by-step interface, summary page, product cards, high-fidelity mockups, and frontend implementation.
	Goal	To simplify the configuration of complex hardware products, linking technical choices, catalog, and commercial request.



OUTCOME

An interactive configurator that simplifies hardware selection, generates a technical summary, and suggests similar products, facilitating the path to a contact request.



PROJECT VALUE

Value Generated

The project transformed a technical hardware selection process into a guided, clear experience oriented toward conversion.

- More accessible hardware configuration
- More readable step-by-step flow
- Link between catalog and commercial inquiry

Skills Applied

The work combined experience design, functional logic definition, and frontend development, taking the configurator from mockup to interactive application.

- UX/UI for a complex technical flow
- Configuration and selection logic
- AI-Assisted Frontend Development

Contribution

Holding full ownership of the entire project, I was able to transform a technical hardware selection process into a guided digital flow, handling UX, functional logic, UI, and a testable frontend prototype before final development.

- Guided flow for hardware configuration
- UX/UI for selection and technical summary
- AI-assisted frontend development

SECO DEVELOPER EXPERIENCE

Project Overview

CONTEXT

SECO needed to improve its Developer Experience, creating a **digital ecosystem** capable of gathering technical documentation, hardware and software resources, product guides, and AI applications in a **clearer path** accessible to clients, developers, and partners.

APPROACH

The work started from the analysis of content and of **the needs of technical users** and touchpoints between hardware, software, and digital services. The design focused on information architecture, navigation paths, resource organization, and interfaces capable of guiding the user from technical consultation to the **discovery** of new solutions.

CHALLENGE

Content and tools were distributed across **different points of the ecosystem**. A **more readable** structure was needed to help developers, clients, and partners find technical information, explore products, evaluate compatibility, and discover software or AI applications linked to SECO hardware.

IMPACT

The project defined a Developer Experience **more organized and scalable**, making documentation, product resources, and applications more accessible. The **new structure** supports technical consultation, onboarding, discovery, lead generation, and possible evolutions toward software activation or deployment flows.

Developer Center

A documentation platform to gather and make accessible technical resources, guides, and hardware and software documentation from the SECO ecosystem, offering a single access point for clients, developers, and partners.



Role

User experience definition, information structure, and design of the platform's main interfaces.



Deliverable

Content map, navigation structure, page templates, product sections, UI components, editorial guidelines, and high-fidelity mockups.



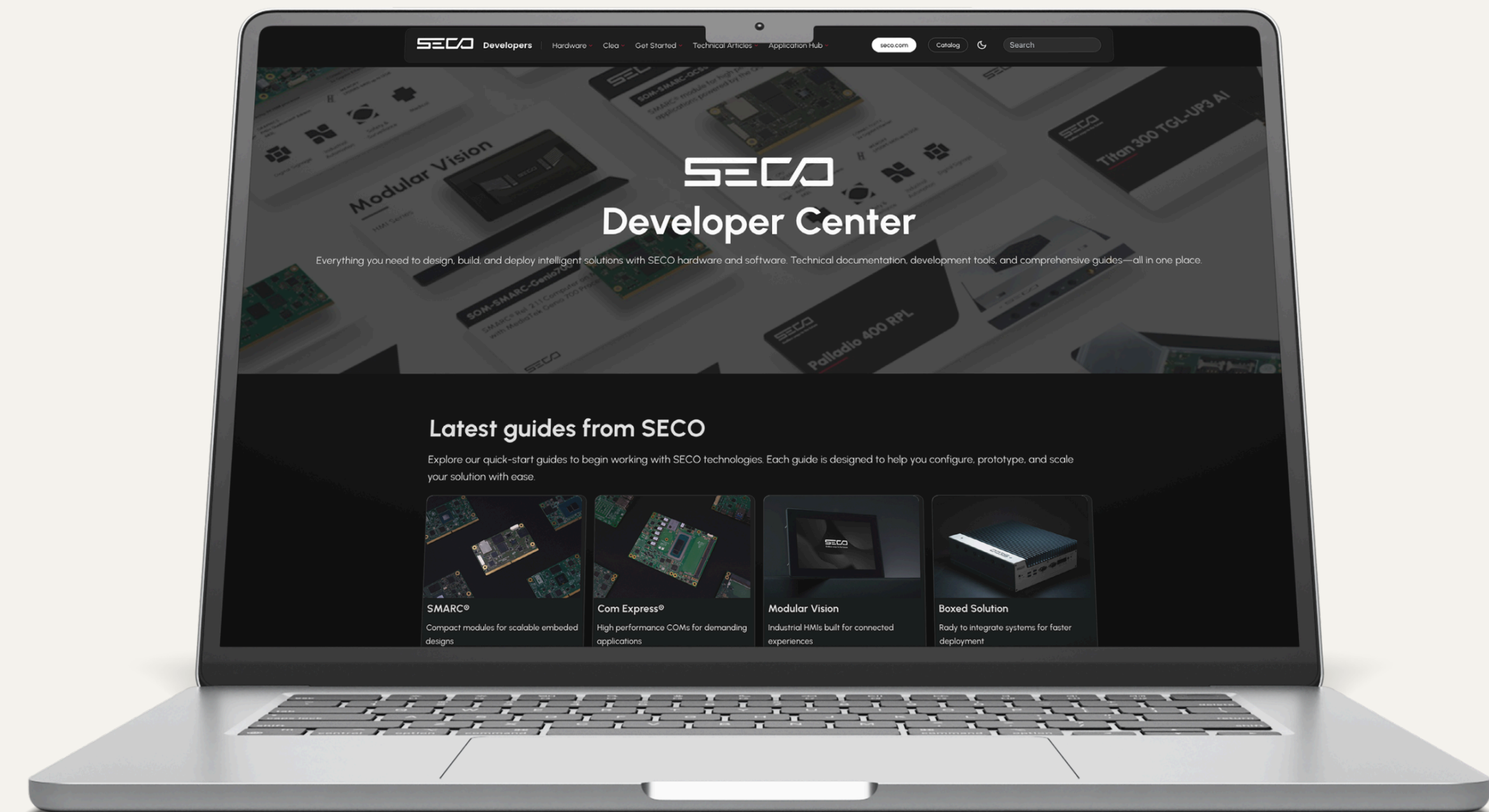
Goal

To create a single documentation hub to simplify access to guides, technical resources, and documentation for the entire hardware and software catalog.



OUTCOME

A clearer and more scalable platform for technical consultation, capable of supporting onboarding, product exploration, and resource access for developers, clients, and partners.



Application Hub

A platform designed to gather and make more accessible AI applications, models, and software solutions compatible with the SECO hardware ecosystem, creating a discovery hub dedicated to developers, clients, and partners.



Role

UX/UI design of the App Hub experience, from defining discovery flows to designing detail pages and contact paths.



Deliverable

User flows, catalog architecture, filters, detail pages, and high-fidelity mockups.



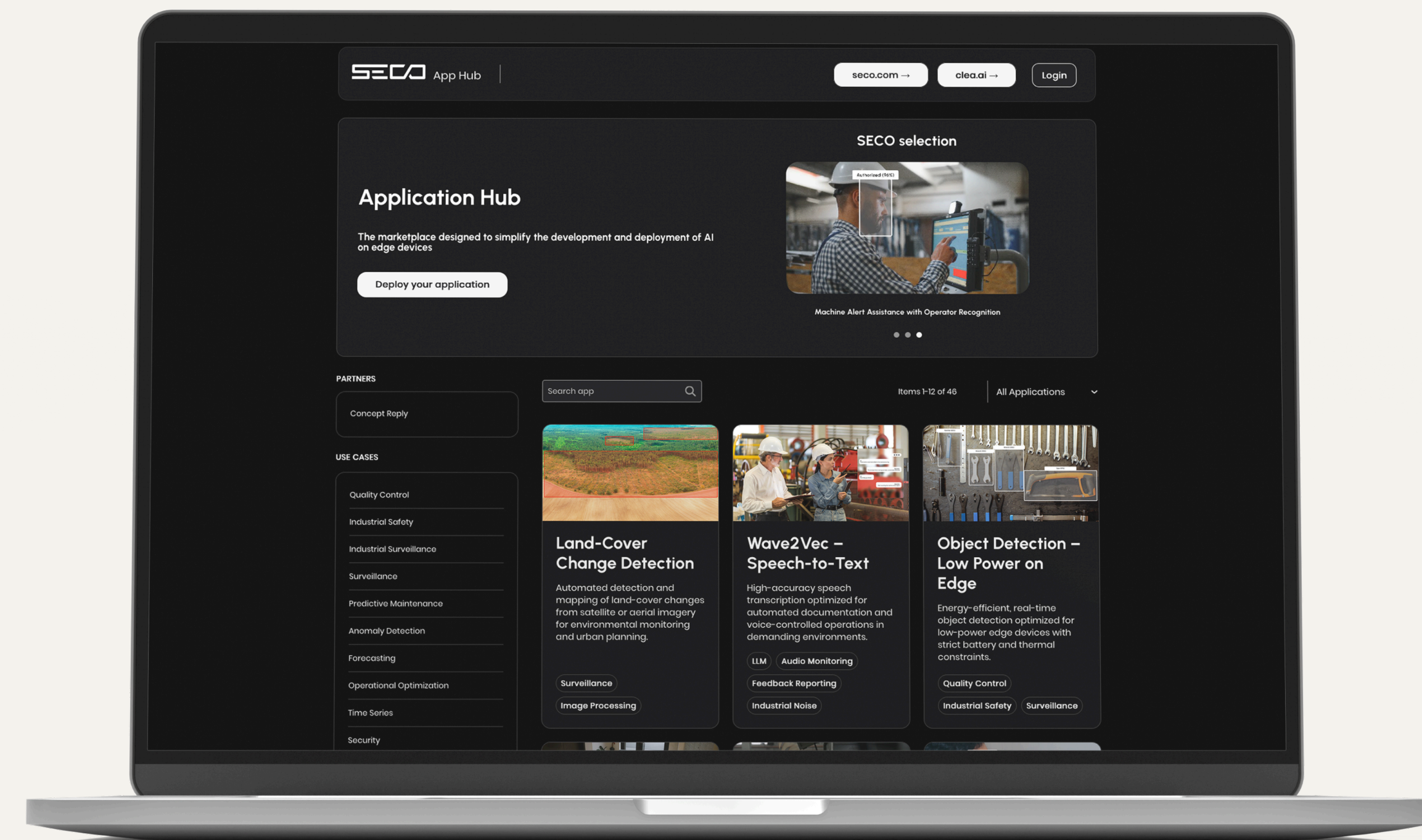
Goal

A digital hub to simplify the discovery of AI applications and software solutions compatible with SECO devices, connecting catalog, technical compatibility, and commercial opportunities.



OUTCOME

A more structured experience for exploring AI and software solutions, capable of supporting discovery, technical evaluation, and lead generation within the SECO Developer Experience.



PROJECT VALUE

Value Generated

The project transformed scattered content, tools, and resources into a clearer, more accessible digital experience oriented around the Developer Experience.

- Centralization of documentation and resources
- Greater continuity between content and paths
- Support for onboarding and consultation

Skills Applied

The project required a strong ability to synthesize technical complexity, user needs, and business objectives.

- Architecture for technical content
- UX/UI for B2B and developer platforms
- Organization of discovery paths

Contribution

I designed the structure of the developer experience, organizing documentation, product resources, and AI applications into clearer paths for consultation, discovery, and commercial contact.

- Technical content structure
- Products and resources documentation hub
- Discovery for apps and AI solutions

THANK YOU

Thank you for your time.
For further information, collaborations,
or new opportunities, I can be reached
at the following contacts.

Federico Baccetti

Senior UX/UI & Service Designer

EMAIL

federico.baccetti92@gmail.com

PHONE

+39 347 0313 500

LINKEDIN

[Federico Baccetti](#)

WEBSITE

www.federicobaccetti.com