

Benoit Meunier

ALIAS_PENDING@benoitmeunier.com | linkedin.com/in/benoit-meunier | github.com/bmeunier1974 |
benoitmeunier.com | Montreal, QC, Canada

PROFESSIONAL SUMMARY

Senior software engineer with 26 years shipping production software, now working at the seam between product engineering and agentic AI systems, with deep interface development expertise and understanding. Sole author of two greenfield enterprise React 19 / TypeScript SPAs: 75+ entity pages, 50+ data grids, and a 100+ component UI kit on a custom state architecture with no external state library. On the back end, a Go service that turns natural-language queries into typed, validated filters, and roughly 140 typed Python modules where an LLM agent drives a curated tool registry and writes code that is validated in a subprocess before it can land. Before that, 12.5 years of carrier-grade set-top box delivery to millions of subscribers, shipping minimal code into fixed memory and render budgets past a carrier's own deployment gate. Bilingual French and English, based in Montreal.

CORE COMPETENCIES

Agentic AI systems

LLM platform / LLMOps

React 19 + TypeScript

Go HTTP services

Typed Python 3.10-3.13

GraphQL / Apollo Client 4

Design systems

MCP SDK

Claude Code

Structural enforcement over convention

Constrained runtimes

Release & supply-chain engineering

WORK EXPERIENCE

Bminty (formerly ProConsultant Informatique)

Mar 2020 - Present

Senior Software Engineer

Montreal, QC

- Sole author of two large-scale enterprise SPAs built from scratch (React 19, TypeScript, GraphQL, Vite, Tailwind 4, Shadcn/Radix, Apollo Client 4), owning every layer from API contract to rendered interface
- Delivered 75+ entity pages, 50+ advanced data grids, and 50+ multi-tab forms with validation, dirty tracking, and optimistic concurrency, plus a proprietary design system of 100+ components with a hot-swappable Dark / Light / Branded theme
- Built services/core/ai-api (Go): turns natural-language queries into typed SearchQueryFilterInput[] arrays, with pluggable LLM and vocabulary backends and a full test pyramid including parity tests
- Designed a custom state architecture of 25+ composable React Context providers with hierarchical dependency gating, fully type-safe, with no external state libraries
- Built a field-level permission system across 60+ business entities so the interface only renders what a given user is cleared to see, plus 50+ locale files across 6 global regions via FormatJS
- Implemented GraphQL subscriptions over WebSockets and dynamic fragment generation tied to visible grid fields, cutting payloads down to the data actually on screen
- Authored tools/Ing-ui-mcp-server (Node + MCP SDK) exposing the internal UI catalog to GitHub Copilot, plus the entire Copilot customization layer used by every engineer on the project

LVL Studio

Oct 2007 - Apr 2020

Senior Software Engineer

Montreal, QC

- Owned end-to-end TV user experience and interface design across set-top box, mobile, and desktop, contributing to nearly every major TV interface the studio produced in that time
- Shipped minimal code into resource-constrained embedded set-top hardware on Microsoft Mediaroom and Bluestreak, where memory and render budgets are fixed and every release passes the carrier's own deployment gate

- Carrier-grade delivery to millions of subscribers at Bell Canada and Telus, plus Time Warner Cable and Hayu

Purelogic

May 2001 - Oct 2007

Software Engineer

Montreal, QC

Sid Lee

Aug 1999 - May 2001

Web Developer

Montreal, QC

- Early career at a leading Montreal creative agency

PROJECTS

Noctis - autonomous agentic research system MIT

An LLM agent drives a formulate / match / optimize / decide cycle through a curated tool registry, authoring one reviewable Python file per candidate and validating it in a subprocess before it can land. Two interchangeable agent drivers behind one interface. ~140 typed modules, 93 test modules, 93% coverage, ports-and-adapters seams keeping every heavy dependency optional. Validation enforced structurally in code: walk-forward splits, out-of-sample on temporal and symbol axes, overfit gap guards, and a tested no-lookahead invariant. Paper-only by construction behind two independent safety gates. Written with Claude Code.

Python - github.com/bmeunier1974/noctis

noctis-ollama - local LLM model plane MIT

One-command provisioning turning a bare GPU machine into a verified, keyless LLM backend an agent loop can drive tools through, gated by a 7-gate contract test asserting exactly what the consuming code parses: tool-call round-trip, OpenAI /v1 parity, and 100% GPU residency. Fit a 14B model with a pinned 16k context into 12 GB VRAM via q8_0 KV-cache quantization and flash attention. Settled a tool-dispatch protocol bug empirically with a LiteLLM parity harness.

Bash + Python - github.com/bmeunier1974/noctis-ollama

lexiqr - deterministic entity-resolution library MIT

Deterministic resolution of per-tenant private vocabulary to canonical database entities for multi-tenant B2B back ends, chosen over embeddings and NER for explainability and a CI-enforced p95 under 10 ms. Semver-governed public API enumerated as a contract, determinism verified across operating systems and Python versions against a committed golden report, 45 test modules against 15 source modules. Tag-to-PyPI release pipeline on OIDC trusted publishing, fronted by a consistency gate (pipeline built and tested; first publish not yet performed).

Python - github.com/bmeunier1974/lexiqr

Grid Manager - visual quantitative strategy builder private

A visual DAG editor for composing, backtesting, and optimising trading strategies, backed by a FastAPI engine and an LLM-assisted generation service. A single authenticated gateway fronts the system; long optimisation runs queue to a worker with progress streaming back to the browser over a WebSocket.

TypeScript, React, Python, FastAPI, PostgreSQL, Redis, Docker

Community Knowledge AI - document-grounded bilingual assistant private

Turns a condo association's document archive (bylaws, minutes, budgets, resolutions) into a bilingual assistant members question in plain language, every answer cited back to its source document. An LLM-maintained knowledge wiki sits between archive and answers, compiled once rather than re-derived per query, so quality compounds as documents accumulate.

TypeScript, Next.js, Claude Agent SDK, PostgreSQL, Docker

Commander - control plane for independent SaaS products private

A control plane for launching and operating independent SaaS products from one place: an admin console, a Supabase-backed platform database, and a scaffolder that generates a new app with its own repository and vendored shared packages. By construction never a runtime or deploy-time dependency of anything it manages.

TypeScript, Next.js, Turborepo, pnpm, Supabase, PostgreSQL

Nursing Course Helper - AI course-design platform private

An AI-assisted platform for nursing instructors, built with a Montreal teaching hospital: plans courses, drafts presentations, and designs assessments against Quebec's MEQ programme competencies and the OIIQ's professional standards, with agents fed by an MCP server serving the pedagogical and regulatory reference data.

JavaScript, Model Context Protocol, Copilot agents & skills

beautify-md - markdown readability skill for Claude Code open source

A Claude Code skill that measures, audits, rewrites, and authors markdown so humans enjoy reading it - six layers, every rule traced to readability research, with a bundled bash and awk script that keeps the numbers honest. No build step, no keys, no external services.

Shell + awk - github.com/bmeunier1974/beautify-md

SKILLS

Frontend: TypeScript, React 18/19, GraphQL, Apollo Client 4, TailwindCSS 4, Radix UI / Shadcn, Vite, React Hook Form, Zod, RxJS, React Intl / FormatJS, CVA design systems

Python: Python 3.10-3.13 (typed), Pydantic 2, Typer, pandas, NumPy, PyArrow / Parquet, rapidfuzz, hatchling, strict mypy, PEP 561 py.typed

Backend & services: Go (HTTP services), Node, REST, GraphQL subscriptions over WebSockets, multi-tenant isolation, field-level permissions

AI / LLM: LLM integration & prompt engineering, Model Context Protocol (MCP) SDK, Anthropic SDK / Claude API, LiteLLM, Ollama & local serving, GitHub Copilot customization, Claude Code

Quantitative / data: NautilusTrader, vectorbt, Optuna, DataBento, exchange-calendars, APScheduler, quantstats, Parquet data lakes

Testing & tooling: pytest, Hypothesis, Vitest, Playwright, Testing Library, ruff, ESLint 9, uv, pre-commit, GitHub Actions matrix CI, shellcheck, Docker Compose, PyPI trusted publishing (OIDC)

Languages: French (native), English (fluent)