w3-suite/

├─ apps/

│ ├─ frontend/

│ │ ├─ web/ # SPA “Suite” (tenant-facing)

│ │ │ ├─ public/

│ │ │ └─ src/

│ │ │ ├─ app/ # bootstrap, router, providers, theme

│ │ │ ├─ layout/ # AppShell/Header/Sidebar (glass)

│ │ │ ├─ features/ # feature-first

│ │ │ │ ├─ cassa/

│ │ │ │ ├─ magazzino/

│ │ │ │ ├─ settings/ # RS, Store, RBAC UI, billing override

│ │ │ │ ├─ crm/ # leads, customers, campagne, inbox

│ │ │ │ ├─ gare/

│ │ │ │ ├─ report/

│ │ │ │ ├─ hr/

│ │ │ │ └─ cms/ # editor vetrina store (landing/form/CTA)

│ │ │ ├─ core/ # api client, access guard, hooks comuni

│ │ │ ├─ styles/ # tailwind.css, tokens bridge

│ │ │ ├─ assets/

│ │ │ └─ tests/

│ │ │ ├─ index.html

│ │ │ ├─ vite.config.ts

│ │ │ ├─ tailwind.config.ts

│ │ │ └─ tsconfig.json

│ │ └─ brand-web/ # SPA “Brand Interface” (HQ)

│ │ ├─ public/

│ │ └─ src/

│ │ ├─ app/ # bootstrap brand, OIDC brand

│ │ ├─ layout/ # shell con settori (marketing/sales/ops/admin)

│ │ ├─ core/ # BrandAccessProvider, guards, scope switcher

│ │ ├─ features/

│ │ │ ├─ sales/ # KPI, forecast (DWH)

│ │ │ ├─ marketing/ # campagne globali, audiences, analytics

│ │ │ ├─ ops/ # inventory rete, alert

│ │ │ ├─ amministrativo/ # billing tenant, flags

│ │ │ └─ dev-tools/ # tab unica Codegen/QA/Docs (agent hub)

│ │ ├─ styles/ └─ tests/

│ │ ├─ index.html ├─ vite.config.ts ├─ tailwind.config.ts └─ tsconfig.json

│ └─ backend/

│ ├─ api/ # NestJS – Suite (OLTP)

│ │ └─ src/

│ │ ├─ core/ # auth (OIDC+MFA), rbac, tenancy (RLS), audit, settings-base

│ │ ├─ db/

│ │ │ ├─ schema/ # Drizzle schema per modulo

│ │ │ ├─ migrations/ # SQL (indici + RLS)

│ │ │ └─ seed/

│ │ └─ modules/ # DDD-lite per dominio

│ │ └─ <modulo>/

│ │ ├─ domain/ # entità, policies, eventi

│ │ ├─ application/ # services/use-cases

│ │ ├─ infra/ # repo, integrazioni, schema.ts del modulo

│ │ └─ presentation/ # controller, dto, validators

│ │ ├─ test/ └─ (main.ts, app.module.ts, tsconfig, jest.config)

│ ├─ brand-api/ # NestJS – Brand (DWH + orchestrazione)

│ │ └─ src/

│ │ ├─ auth/ ├─ rbac/ # ruoli per settore + scope rete

│ │ ├─ analytics/ # query DWH (read-only)

│ │ ├─ pricing/ # listini brand (versioni)

│ │ ├─ campaigns/ # campagne brand → propagation

│ │ ├─ templates/ # WABA/Email/SMS templates globali

│ │ ├─ propagation/ # emitter eventi (BullMQ) → workers

│ │ └─ shared/ # DwhService, QueueService, utils

│ │ ├─ test/ └─ (main.ts, app.module.ts, tsconfig, jest.config)

│ ├─ workers/

│ │ └─ brand-propagation/ # consumer eventi brand→tenant (BullMQ)

│ │ ├─ src/ (queues, jobs, handlers, security)

│ │ └─ tsconfig.json

│ └─ cms-render/ # (opz) renderer edge (Astro/Workers) per landing pubbliche

│ ├─ src/ ├─ astro.config.mjs └─ tsconfig.json

├─ packages/

│ ├─ ui/ # design system (Tailwind+shadcn, glass)

│ │ └─ src/ (Button, Card, Table, Sidebar, Header, OfferCard, PillFilter, CTA…)

│ ├─ tokens/ # Tailwind preset + CSS vars (palette W3: arancio/viola/nero)

│ │ ├─ tailwind-preset.ts └─ index.css

│ ├─ sdk/ # client TS: sdk.api (suite) + sdk.brand (brand-api)

│ │ ├─ api/ (OpenAPI client o fetch wrapper)

│ │ └─ brand/

│ ├─ dwh/ # query/typing DWH (zod/types, sql tagged)

│ ├─ cms-core/ # schema/logic CMS (sites/pages/forms, publish)

│ ├─ cms-render/ # blocchi di rendering condivisi (Astro/React)

│ └─ agents/ # strumenti AI per dev (PR-only)

│ ├─ runner/ # codegen/qa/docs → crea PR

│ ├─ adapters/ # integrazione replit/lovable/… (opz)

│ ├─ orchestrator/ # lib per split task + gh client

│ └─ prompts/ # prompt templates & convenzioni

├─ db/

│ ├─ oltp/ # migrazioni OLTP (Suite)

│ ├─ brand/ # migrazioni Brand API (cataloghi, deployments)

│ └─ dwh/ # schema DWH (read-only)

├─ configs/ # preset condivisi

│ ├─ eslint/ .eslintrc.base.cjs

│ ├─ prettier/.prettierrc

│ ├─ tsconfig/ tsconfig.base.json

│ ├─ vite/ vite.base.ts

│ └─ jest/ jest.base.cjs

├─ .github/

│ └─ workflows/

│ ├─ ci-frontend-web.yml

│ ├─ ci-frontend-brand.yml

│ ├─ ci-backend-api.yml

│ ├─ ci-backend-brand.yml

│ ├─ ci-workers.yml

│ ├─ agent-orchestrator.yml # Codegen → QA → Docs (OpenAI orchestrator)

│ └─ release.yml

├─ docker/

│ ├─ docker-compose.dev.yml # postgres, redis, keycloak/minio

│ ├─ api.Dockerfile

│ ├─ brand-api.Dockerfile

│ ├─ web.Dockerfile

│ ├─ brand-web.Dockerfile

│ ├─ worker.Dockerfile

│ └─ cms-render.Dockerfile

├─ scripts/ # DX & ops

│ ├─ db.reset.ts / db.seed.ts

│ ├─ codegen.ts (openapi) / agents.ts

│ └─ dwh.sync.ts (ETL semplice)

├─ docs/

│ ├─ ADR/ # decisioni architetturali

│ └─ ARCHITECTURE.md # overview generali

├─ pnpm-workspace.yaml

├─ turbo.json # (opz) orchestratore build

├─ tsconfig.base.json

├─ drizzle.config.ts # multi-project (oltp/brand/dwh)

├─ package.json

└─ README.md