

# NEXUS PRIME

MVP UI Preview (spec-aligned, private-by-default)

- Login + JWT token storage (localStorage)
- Applications list with quick-create flow
- Dark-mode dashboard shell (React + Tailwind)
- Designed for Docker private localhost deployment

# NEXUS PRIME

Adaptive AI Integration Platform (MVP)

## Sign in

Email

Password

**Login**

Default bootstrap credentials come from API env vars.

### What this screen does

- Calls POST /v1/auth/login
- Stores access\_token in localStorage as np\_token
- Redirects into Applications list on success

Demo credentials (from README):

**admin@nexus.local / adminadmin**

## Applications

0 total

No applications yet. Click “New Application”.

### Example: Intent Ingestion Layer

Governed intent objects for automation reliability

Type: saas • Status: ready • Learning: 0%

Endpoints: 0

Tables: 0

Placeholder card shown for context in this preview (real app starts empty).

## Applications

3 total

### Checkout Intelligence

Tracks conversion events + latency for commerce flows

Type: saas • Status: learning • Learning: 42%

Endpoints: 18

Tables: 3

### Support Triage Connector

Routes support intent to the right workflow with audit trails

Type: support • Status: ready • Learning: 18%

Endpoints: 9

Tables: 1

### Observability Bridge

Discovers endpoints (OpenAPI) and produces baseline insights

Type: api • Status: ready • Learning: 65%

Endpoints: 27

Tables: 5

### Quick-create flow

In the MVP UI, “New Application” prompts for a name and posts to POST /v1/applications.

# Run the real interactive demo

## Option A — Docker (as README suggests)

From the provided repository (private localhost).

- `cd infra`
- `docker compose up --build`
- Open Web UI: <http://localhost:5173>
- API docs: <http://localhost:8000/docs>

## If running without Docker

- Update Vite proxy target in `apps/web/vite.config.ts` from `http://api:8000` to `http://localhost:8000`
- Set API `DATABASE_URL` to a local Postgres or a SQLite URL (requires `aiosqlite`) for dev
- Then run: (api) `uvicorn app.main:app --reload` and (web) `npm run dev`

Note: This environment can't fetch npm/pip dependencies, so the PDF above is a faithful UI mock based on source code.