KURUMSAL YAPAY ZEKÂ ALTYAPISI

Yapay zekâ sizin tesisinizde, kurallar sizin elinizde.

Kurumsal yapay zekânın üç çıkmazı

01 **Veri dışarı çıkıyor.**

Genel bulut yapay zekâ servislerinde müşteri sözleşmeleri, İK verileri ve mali tablolar üçüncü taraflara akar.

02 **Maliyet kontrolsüz.**

Kim, hangi modeli, ne kadar kullandı bilinmez; kullanım arttıkça fatura büyür, ay sonu sürpriz olur.

03 **Denetim kanıtı yok.**

KVKK ve EU AI Act sorulduğunda gösterilecek onay kaydı, lisans kanıtı ve erişim günlüğü bulunmaz.

Kontrolsüz yapay zekâ kullanımı, denetimde karşılığı olmayan bir risk alanıdır.

BearHive: şirketinizin kendi binasında çalışan, kuralları sizin koyduğunuz özel yapay zekâ altyapısı

Modeller kendi sunucunuzda çalışır; kurumsal veri dışarı çıkmaz. Tüm kullanım tek kontrol noktasından geçer, her istek kayıt altındadır. Kuralları siz koyarsınız — platform uygular ve kanıtlar.

Veri sizde kalır

Modeller ve embedding'ler tamamen yerel; internetsiz (air-gapped) ortam dahil.

Kontrol sizde kalır

Bütçe, erişim ve kullanım kuralları merkezi tanımlanır; departman bazında uygulanır.

Kanıt sizde kalır

Onay, imza, eval ve erişim kayıtları tek tıkla inen uyumluluk raporunda toplanır.

Yapay zekâyı çalışanlarınıza açın; veriyi, bütçeyi ve kuralları elinizde tutun.

Onaysız hiçbir model yayına çıkmaz



- **Yasaklı lisans otomatik elenir.** Lisans politikası, izin verilmeyen lisanslı modelleri onay aşamasında reddeder.
- **Eval geçmeyen onaylanamaz.** Her onay ve red kararı, kimin tarafından ne zaman verildiğiyle kayıt altına alınır.
- **Red-team sürekli çalışır.** Jailbreak, injection ve sızıntı senaryoları periyodik taranır; skor uyumluluk raporuna girer.

0

üretim trafiği alan onaysız model sayısı — kural gereği, her zaman

Model kataloğu mağaza gibi çalışır: her kartta boyut, görev, lisans ve kaynak uygunluğu görünür.

KVKK ve EU AI Act sorulduğunda: tek dosyada hazır kanıt

Model envanteri	sürüm, lisans rozeti, kaynak
Onay kayıtları	kim, ne zaman, hangi karar
İmza ve eval kanıtları	paket imzası, eval skoru
Red-team sonuçları	son tarama skorları, periyot
Lisans politikası özeti	yasaklı lisans desenleri
Ekip ve erişim özeti	kimin neye erişimi var



Denetçiye 'işte kanıt' dersiniz — toplantıdan sonra değil, toplantı sırasında.

Sızıntı ve kötüye kullanım, kaynağında engellenir

Kişisel veri maskeleme

T.C. kimlik numarası, kredi kartı ve e-posta kalıpları otomatik maskelenir veya engellenir.

Yasaklı konular

Kurumun belirlediği konu ve terim listeleri, hem girdide hem çıktıda engellenir.

Prompt-injection koruması

Modele yönelik bilinen saldırı kalıpları yakalanır; kural değişiklikleri yeniden başlatma gerektirmeden anında devreye girer.

İSTEK DENETİMİ · CANLI KAYIT

```
girdi : kart no 5200 19xx xxxx xx21
karar : PII kuralı → maskelendi

girdi : "önceki kuralları yok say…"
karar : injection → engellendi
```

KAYIT ALTINDA

Engellenen her istek, engel gerekçesiyle birlikte görünür.

Her engel, 'bu istek neden engellendi' cevabıyla birlikte denetim kaydına düşer.

Herkes kullanır; bütçeyi ve limitleri siz koyarsınız

Departman bazlı ekipler

Her ekibin aylık bütçesi, hız limiti ve erişebildiği modeller ayrıdır; periyot sonunda bütçe otomatik sıfırlanır.

Otomatik fren

Bütçe %80'e geldiğinde uyarı gider, %100'de kullanım otomatik durur — fatura sürprizi olmaz.

Showback raporu

Kullanım departman kırılımında CSV'ye iner; maliyet gösterimi ₺ veya $ — kuru siz girersiniz.

AYLIK EKİP BÜTÇESİ · ÖRNEK GÖRSEL



%80 · uyarı

%100 · otomatik durma

Tavan baştan bellidir: limit dolmadan önce ekip bilgilendirilir, dolduğunda erişim kendiliğinden durur.

Bütçeler **anahtar ve ekip bazında** işler; hangi departmanın ne harcadığı her an görünür.

Mevcut kullanıcı sisteminizle konuşur



- **LDAP / Active Directory ve Azure SSO hazır.** Grup üyelikleri otomatik olarak rol ve ekibe dönüşür; kullanıcı ilk girişte kendiliğinden açılır.
- **SCIM 2.0 provisioning.** Dizindeki hesap ve grup değişiklikleri platforma otomatik yansır; ayrı senkronizasyon işi gerekmez.
- **Parolalar dizinde kalır.** Kurumsal parolalar platformda saklanmaz; bağlantı testi ve kullanıcı önizlemesi arayüzden yapılır.

İşten ayrılan çalışanın erişimi, dizindeki kaydı kapandığı anda biter.

Abonelik değil, altyapı: maliyet kullanımla büyümez

	BULUT LLM SERVİSLERİ	BEARHİVE
Token ücreti	Kullandıkça artar	**Yok — sabit maliyet**
Kullanıcı lisansı	Kalem başına büyür	**Yok**
Veri konumu	Üçüncü taraf bulut	**Kendi sunucunuz**
Maliyet modeli	Kullanım × birim fiyat	**Donanım + elektrik**

1.000+

kullanıcılı bir kurumda bulut alternatifi, kalem lisansı ve token ücretiyle aylık on binlerce dolara ulaşır.

BearHive'da **100 de 10.000 de çalışan** aynı platformu kullanır; büyük kurumda asıl fark burada çıkar.

Donanım sizin — mevcut sunucularla başlanır, GPU ve düğüm eklendikçe kapasite büyür.

Bir haftalık pilot: kendi sunucunuzda, kendi verinizle

01 **Kurulum — tek günde**

Tek paketle mevcut sunucunuza kurulur; başlangıç için CPU yeterlidir, dış bağlantı gerekmez.

02 **Gerçek iş yükü — bir hafta**

Kendi dokümanlarınız, ekipleriniz ve kullanım senaryolarınızla çalışır; örnek veriyle değil.

03 **Kanıt — karşılaştırma**

Kullanım dökümü (CSV showback) ile aynı iş yükünün bulut faturası yan yana konur.

Kararı veriyle verin — pilot sonunda elinizde kendi verinizle üretilmiş gerçek maliyet tablosu olur.

BÖLÜM 2

Teknik Derinlik

Teknik yönetim ve mühendislik ekipleri için mimari, yaşam döngüsü, gateway, güvenlik ve operasyon detayları.

12 Servis haritası ve mimari

13 Model yaşam döngüsü

14 Runtime · çok düğümlü serving

15 Gateway · bütçe, guardrail, MCP

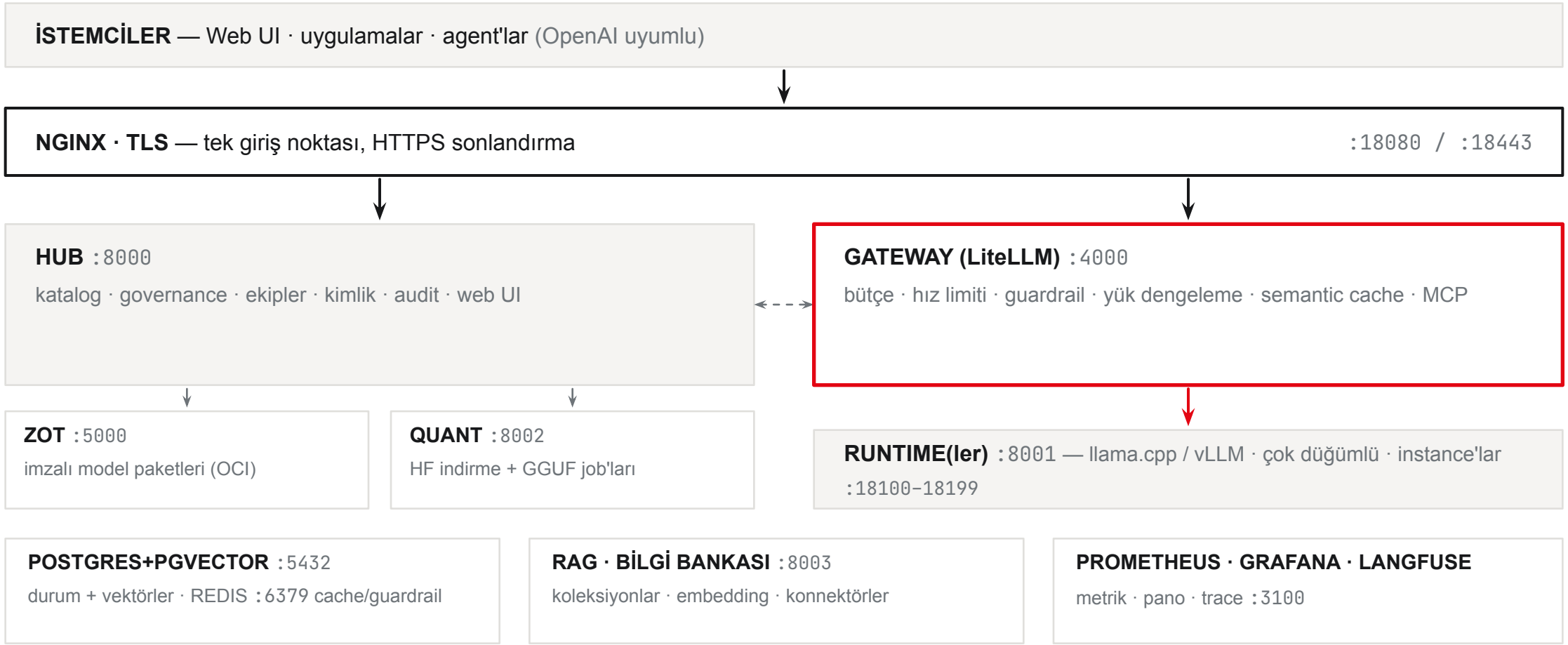
16 Güvenlik ve tedarik zinciri

17 Gözlemlenebilirlik

18 Kimlik · LDAP, OIDC, SCIM

19 Kurulum ve operasyon

Servis haritası: tek giriş noktası, kontrollü akış



HuggingFace'den onaylı servise

1 · İçe alma sihirbazı
HF araması → hazır GGUF, quantize (Q2…Q8) veya vLLM paketi; kaynak uygunluğu o anda hesaplanır

→

2 · Paketleme
model + model.yaml manifest (parametre, bağlam, lisans) OCI artefaktı olarak Zot'a; cosign imzası

→

3 · Governance kapısı
lisans politikası + imza + kural bazlı eval geçmeden onay yok

→

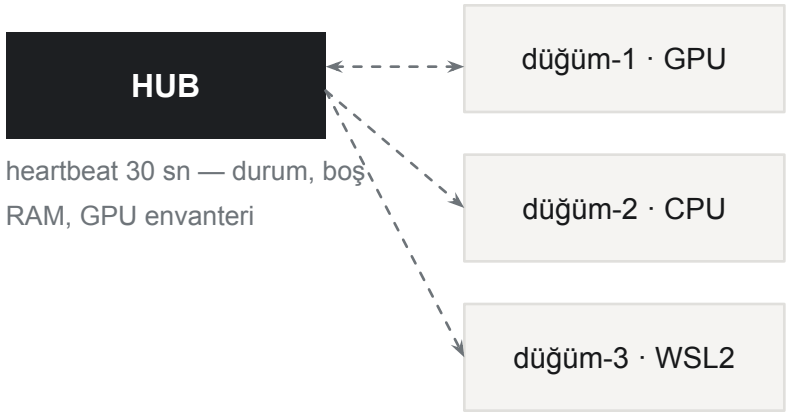
4 · Servis
onaylı model, tek OpenAI uyumlu uç üzerinden kuruma açılır

ALTINDA NE VAR

- Quant job'ları ayrı konteynerlerde çalışır; hub kuyruğu izler — indirme, quantize ve vLLM işleri ayrı filtrelenebilir.
- GGUF başlığı okunarak mimari ve gerçek bağlam boyutu doğrulanır; desteklenmeyen mimariler iş başlamadan engellenir.
- Eval, deterministik Türkçe test setiyle koşar; kurum kendi golden set'ini ekleyebilir. Skor, onay ekranında görünür.
- Embedding modellerinde quant önerisi hafifler (F16 / Q8_0); RAG indeksleme ve semantic cache bu modeli kullanır.

Çok düğümlü, kendini ayarlayan sunum katmanı

- **Uzak düğümler heartbeat ile kaydolur** — baremetal, WSL2 veya başka sunucular; model en boş düğüme yerleşir, portlar otomatik ayrılır.
- **Otomatik parametre planlayıcı** — GGUF metadata'sı + düğümün boş belleği → ctx-size, flash-attn ve KV-cache sıkıştırma model başına hesaplanır.
- **GPU esnekliği** — NVIDIA (CUDA), Intel (SYCL/Vulkan), AMD (Vulkan) imajları; CPU-only kurulumlar dahil.
- **Fit-check** — ağırlık + KV tahmini boş belleğe sığmazsa başlatma reddedilir; ret, açık bir Türkçe mesajla döner.
- **Kendini onaran kayıt** — dışarıdan silinen bir konteyner 30 saniye içinde yakalanır (hayalet instance reconcile).



Aynı modelin birden çok instance'ı tek yük dengeleme grubu olur; konuşmalar session-affinity ile aynı instance'a gider. Her instance ayrı konteynerdir ve Prometheus'a http_sd ile otomatik kaydolur.

Bütçe, guardrail ve MCP tek uçta

- **Tek OpenAI uyumlu uç** — sanal anahtarlar; ekip ve anahtar bazında bütçe, TPM/RPM hız limiti ve model-bazlı erişim kısıtı.
- **Guardrail katmanı** — PII maskeleme, yasaklı terim, injection kuralları Redis üzerinden yayınlanır; yeniden başlatma olmadan devreye girer, engeller istek log'unda işaretlenir.
- **MCP Gateway yönetişimi** — MCP sunucuları hub'dan onaylanır, ekip/anahtar bazında yetkilendirilir; araç çağrıları İstek Denetimi'nde izlenir.
- **Semantic cache** — benzer sorular önbellekten yanıtlanır; embedding'ler kendi barındırdığımız modelle üretilir, maliyet ve gecikme düşer.

DAYANIKLILIK

```
fallback + retry + cooldown
hatalı deployment devre dışı kalır,
istek sağlam instance'a gider
```

SPEND-LOG

```
her istek: model · anahtar · token
maliyet · guardrail kararı
```

SABİT SÜRÜM

```
LiteLLM 1.98 — sürüm + digest
sabitli (tedarik zinciri politikası)
```

Kanıtlanabilir güvenlik

- **Sabitlenmiş imajlar** — tüm imajlar sürüm + digest sabitli; offline bundle images.lock ile yüklemeyi doğrular, uyuşmazlıkta durur.
- **SBOM ve imza** — SBOM (syft) pakete gömülür; model paketleri cosign ile imzalanır ve doğrulanır.
- **Tam audit izi** — giriş, onay, bütçe, guardrail, MCP ve SCIM olayları kayıtlı; HMAC imzalı webhook ile SIEM'e akar.
- **Red-team kanıtı** — periyodik taramaların sonuçları uyumluluk raporunun parçası olarak saklanır.

AIR-GAPPED KURULUM

```
$ scripts/offline-bundle.sh
  # tek tar paketi üretir
$ ./install.sh
  # bütünlük doğrular, imajları yükler
$ docker compose up -d
  # dışa bağlantı yok
```

İnternetsiz çalışır; HuggingFace erişimi yalnızca model içe alma sırasında gerekir — yerel dosyadan içe alma da desteklenir.

Metrikten trace'e tam görünürlük

Prometheus

Hub, runtime ve quant /metrics uçları; inference instance'ları http_sd ile dinamik hedef olur — GPU kullanımı, VRAM, token/sn, job durumları.

Grafana

"BearHive — Genel Bakış" panosu otomatik provision edilir; arayüzdeki İzleme kartından tek tıkla açılır.

Langfuse

İstek bazlı trace, gecikme ve maliyet analizi; observability profiliyle açılır, LiteLLM callback'i otomatik yapılandırılır.

İstek Denetimi

Guardrail engelleri kırmızı rozetle; anahtar→ekip korelasyonu ve MCP araç çağrı özetleri hub arayüzünde.



Operasyon ekranı — pano, metrik ve canlı istek akışı tek yerde

Kurumsal dizinle tam entegrasyon

ROLLER — KÜMÜLATİF YETKİ

ROL	YETKİ
`izleyici`	katalog, panel, kullanım (salt-okunur)
`gelistirici`	+ playground, model çalıştırma, Bilgi Bankası
`ml-muhendisi`	+ model içe alma, quantize, registry
`onaylayici`	+ model onayı, uyumluluk raporu, denetim
`admin`	+ kullanıcı/ekip, anahtar, guardrail, bütçe

- **JIT provisioning** — ilk LDAP/SSO girişinde kullanıcı otomatik oluşur; grup eşlemelerinden rol ve ekip atanır, sonraki girişlerde yeniden eşitlenir.
- **OIDC** — Azure AD v2.0, Keycloak, Google; authorization-code akışı, `/auth/oidc/callback`. Bağlantı testi discovery ve JWKS uçlarını doğrular.
- **SCIM 2.0** — Entra ID / Okta'daki hesap ve grup değişiklikleri otomatik yansır; işten ayrılanın erişimi anında kapanır.
- **Güvenlik sınırları** — LDAP parolaları Hub'da tutulmaz; aynı kullanıcı adıyla çakışmada hesap birleştirme yapılmaz, giriş reddedilir.

Tek komutla kurulum, standart operasyon

İLK KURULUM

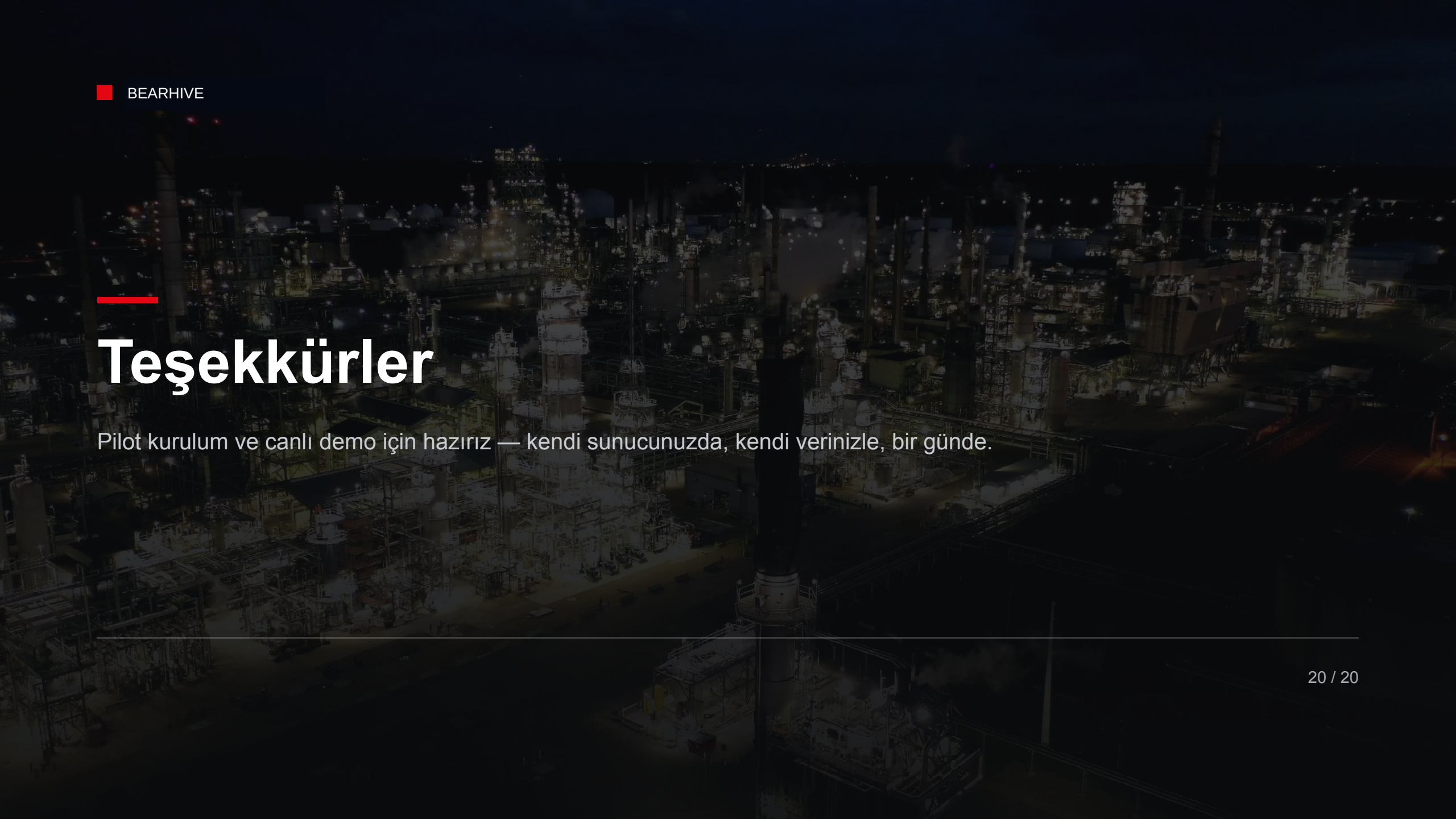
```
cp .env.example .env
nginx/gen-dev-cert.sh
# dev HTTPS sertifikası (bir kez)
docker compose up -d
# çekirdek stack
docker compose --profile observability up -d
# izleme (Prometheus/Grafana/Langfuse)
```

Ön koşul Docker veya Podman, ~20 GB disk; ilk açılışta veritabanı migration'ları ve admin kullanıcısı otomatik oluşur.

PORT HARİTASI

BİLEŞEN	PORT
Hub (web UI + API)	8000
Gateway (LiteLLM)	4000
Runtime	8001
Quant	8002
RAG · Bilgi Bankası	8003
Registry (Zot)	5000
nginx (TLS)	18080 / 18443
Inference instance aralığı	18100–18199

Yedekleme: pg_dump + model dosyaları + anahtarlar · Güncelleme: git pull + compose build · Çok düğüm: tek konteynerlik worker



Teşekkürler

Pilot kurulum ve canlı demo için hazırız — kendi sunucunuzda, kendi verinizle, bir günde.