

Bilko Infrastruktura — objašnjeno jednostavno (za CEO) — 2026-06-09

Bilko Infrastruktura — objašnjeno jednostavno (za CEO)

Verzija 2026-06-09. Cilj: da za 5 minuta razumiješ gdje Bilko živi, kako kod dođe do kupaca, šta je polomljeno i kako to zaobilazimo. Sve provjereno uživo (gcloud), ne napamet.

KORAK 1 — Šta gdje radi (okruženja)

Bilko ima 7 Cloud Run "mašina", ali realno **samo 2 okruženja**:

```
PROD (ono što kupci vide – pravi proizvod, čist, bez demo podataka)
  app.bilko.cloud      -> bilko-web-demo    (web, Next.js)
  app-api.bilko.cloud -> bilko-api-demo  (API, Kotlin/Ktor)
                        |
                        bilko-demo-db      (Postgres 16)

STAGE (za testiranje prije prodaja – UAT)
(interni URL-ovi)     -> bilko-web-stage + bilko-api-stage
                        |
                        bilko-staging-db
```

?? Zašto je zbunjujuće: IMENA LAŽU

Servisi se zovu `...-demo` ali su **to PROD**. Razlog: odlučili smo "reuse demo kao prod" da ne plaćamo novu bazu/servise (0 KM dodatno). Pa `app.bilko.cloud` (tvoj pravi proizvod) radi na mašini

koja se istorijski zove `bilko-web-demo`. Ime = ostatak iz prošlosti, ne realnost.

Ostalo (nije bitno za sad)

- `bilko-demo.alai.no` — stari demo URL, sad samo alias na istu PROD mašinu
- `bilko-web` — prazan stub (ništa)
- `bilko-intesa-demo` — odvojen Intesa (banka) demo
- `bilko-unleash` — servis za feature-flagove
- `bilko-db` — nova prazna prod baza, NE koristi se (prod radi na `bilko-demo-db`)

KORAK 2 — Kako kod do?e iz editora na te mašine (deploy)

Postoje **tri odvojena puta** kako kod ide live, zavisno šta se mijenja:

A) Glavna aplikacija (web + API)

```
ti/agent napišeš kod -> PR na GitHub -> merge u 'main'
                        |
                        v
                GitHub javi Google Cloud Build-u (preko "konekcije")
                        |
                        v
                Cloud Build sagradi Docker image + prođe GATE-ove (security, testovi)
                        |
                STAGE: push u 'main' -> trigger 'bilko-stage-auto-deploy' -> stage mašine
                PROD:  semver tag (npr. v0.2.66) -> trigger 'bilko-main-deploy' -> prod mašine
```

Gate-ovi = automatske provjere prije nego se išta pusti: security skener (Trivy), testovi, lint. Ako bilo koji padne — deploy se zaustavi. To je dobro (štiti prod), ali znači da jedan pokvaren gate blokira SVE.

B) Landing stranice (marketing: `bilko.cloud`, `bilko.io`, `bilko.company`)

```
to su STATIČNE stranice na Cloudflare Pages (ne Cloud Run)
bilko.cloud      = apps/landing-hr
bilko.io         = apps/landing-io
bilko.company    = apps/landing-ba
deploy: GitHub workflow (sad preko FORGE self-hosted runner-a, vidi Korak 3)
```

C) Baza (migracije)

Promjene strukture baze idu kroz Flyway migracije (V1, V2, ... fajlovi). Pokreću se automatski pri deploy-u API-ja. Nikad se stara migracija ne mijenja — uvijek nova.

KORAK 3 — Šta je POLOMLJENO i kako zaobilazimo (stanje 2026-06-09)

Problem	Šta znači	Zaobilaznica	Pravi fix (treba CEO)
GitHub→CloudBuild konekcija mrtva (#103276)	Kad mergeš u main, Google ne dobije signal -> deploy se NE pokrene sam	Deploy ručno: <code>gcloud builds submit</code> (upload koda direktno, preskače mrtvu konekciju, ali vrti sve gate-ove)	Popraviti konekciju treba <code>cloudbuild.connections.update</code> IAM dozvolu — tvoj nalog
GitHub Actions istrošen (#103304)	Privatni repo ima 2000 besplatnih min/mjesec; potrošeno -> CI + landing deploy padaju	FORGE self-hosted runner (tvoj Mac Studio, 10.0.0.2) = besplatno, neograničeno. Već radi za landinge.	Ništa nužno — resetuje se sljedeći mjesec; ili ostavi na FORGE-u zauvijek
Trivy security gate (#103315)	Našao HIGH CVE u Netty biblioteci -> blokirao sve deploje od 02:00	Витрали Netty 4.2.13 -> 4.2.15 (riješeno)	Riješeno ✓
Orphan OAuth brand (#103296)	Stari Google OAuth brand ruši "tag" operacije na bilko-web-demo prometu	Ne koristiti tagove na bilko-web-demo prometu; <code>--clear-tags</code> kad treba	Obrisati brand 762788903040 (Google support)

Pouka: deploy pipeline ima nekoliko slomljenih karika koje su se nagomilale. Glavna (GitHub konekcija) treba tvoju IAM dozvolu. Dok se ne sredi, deploy je ručni preko `gcloud builds submit`.

KORAK 4 — Kako se korisnici LOGUJU (Entra-only)

Bilko više NEMA email+lozinka login. Sve ide preko **Microsoft Entra External ID (CIAM)**:

- Korisnik ode na `app.bilko.cloud/login` -> Microsoft stranica -> upiše email -> dobije **OTP kod na mail** -> uđe
- Prvi put = automatski mu se napravi prazan "tenant" (njegova firma) + 7-dnevni trial
- Stari seed useri (accountant@bilko.ba itd.) su imali lozinku -> **ne mogu se više logovati** (zato smo ih očistili iz baze)

KORAK 5 — Baze i ?iš?enje (2026-06-09)

Prod baza (`bilko-demo-db`) je očišćena: sa **167 firmi / 173 korisnika** -> **2 firme / 3 korisnika**.
Obrisano: 161 test-artefakt + 3 mrtva seed orga. Tvoj nalog `alem@alai.no` migriran na čistu firmu "ALAI Holding AS". Backup napravljen prije (`1781031619853`) — rollback uvijek moguć.

KORAK 6 — Kako TI testiraš

- **Čist proizvod:** `app.bilko.cloud` -> login kao `alem@alai.no` (Entra OTP) -> vidiš prazan, čist Bilko
 - **Seed demo (sa podacima):** u izradi — fiksni nalog `demo@alai.no` + tenant napunjen demo podacima (fakture, kontakti)
-

Vezani MC taskovi

#103300 (prod topologija), #103276 (GitHub konekcija), #103304 (GitHub Actions billing), #103296 (OAuth brand), #103315 (Netty CVE), #103313 (CEO bug-bash), #103308 (landing deploy-dir).

Revision #1

Created 2026-06-09 20:22:25 UTC by John

Updated 2026-06-09 20:22:25 UTC by John