

Arhitektura AI4Scale Platforme

Dino Aljević
FIDIT, Sveučilište u Rijeci
Rijeka, 09.03.2026.



Sadržaj

- 1) Što je AI4Scale?
- 2) Pregled platforme
- 3) Arhitektura
 - 1) Pregled
 - 2) Gateway
 - 3) JupyterHub
 - 4) Jupyter Lab
- 4) Hardware poslužitelja
- 5) Što slijedi?
- 6) Pitanja

Što je AI4Scale?

- AI4Scale je kolaborativna platforma za AI eksperimentaciju namijenjena javnom sektoru, izgrađena kao jedan od tri središnja stupa ekosustava AI4Gov-X.
- Korisnicima omogućuje izgradnju, testiranje i validaciju ML modela u kontroliranom okruženju.
- Platforma podržava federirane identitete, omogućujući korisnicima iz partnerskih institucija pristup putem njihovog postojećeg sustava za prijavu (SSO).

Pregled platforme

- Web aplikacija – prijava putem institucionalnog ili lokalnog računa.
- Popis projekata - stvaranje ili pristup projektima, svaki je izolirani radni prostor.
- Sandbox okruženje - jedan aktivan sandbox po projektu - pokreće JupyterLab instancu.
- Kolaborativno okruženje - uređivanje bilježnica u stvarnom vremenu + asinkroni chat.

Arhitektura - Pregled

Korisnici / Preglednik

↓ HTTPS

Gateway

↓ REST API

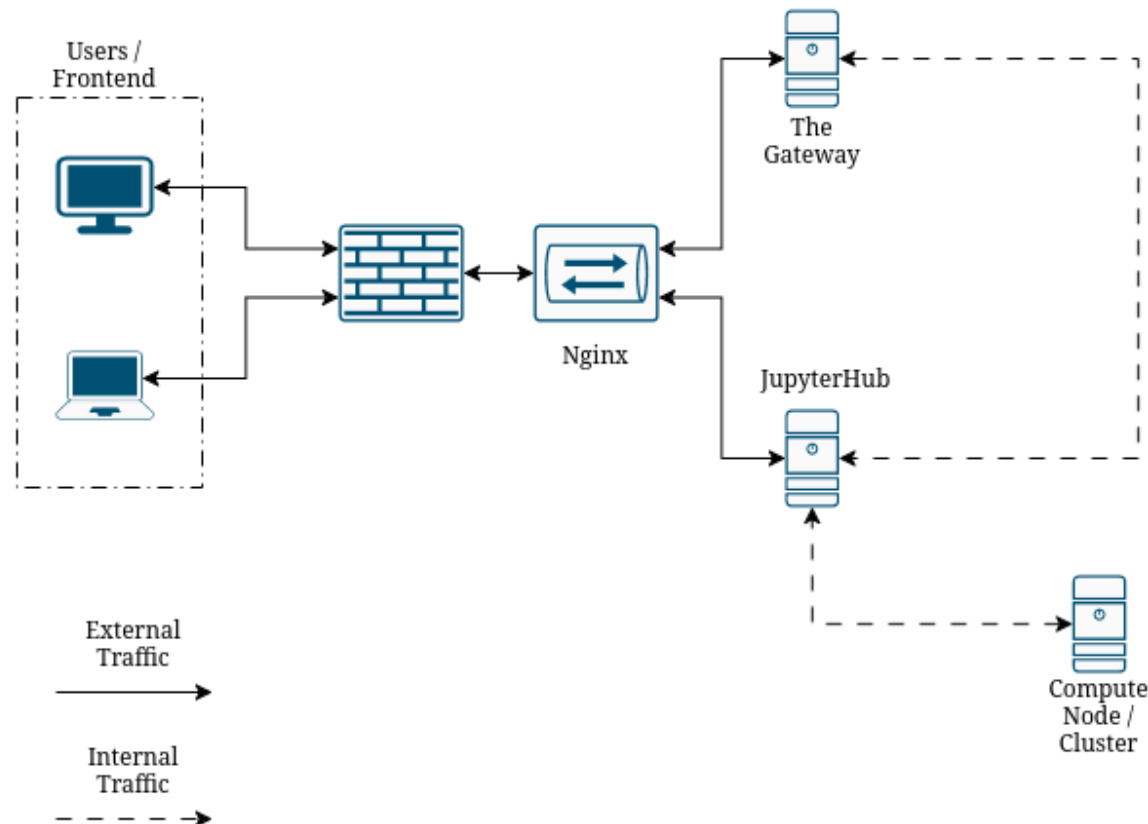
JupyterHub

↓ Spawner (Docker / Kubernetes)

JupyterLab

↑ GPU / CPU / Pohrana

Računalni čvor



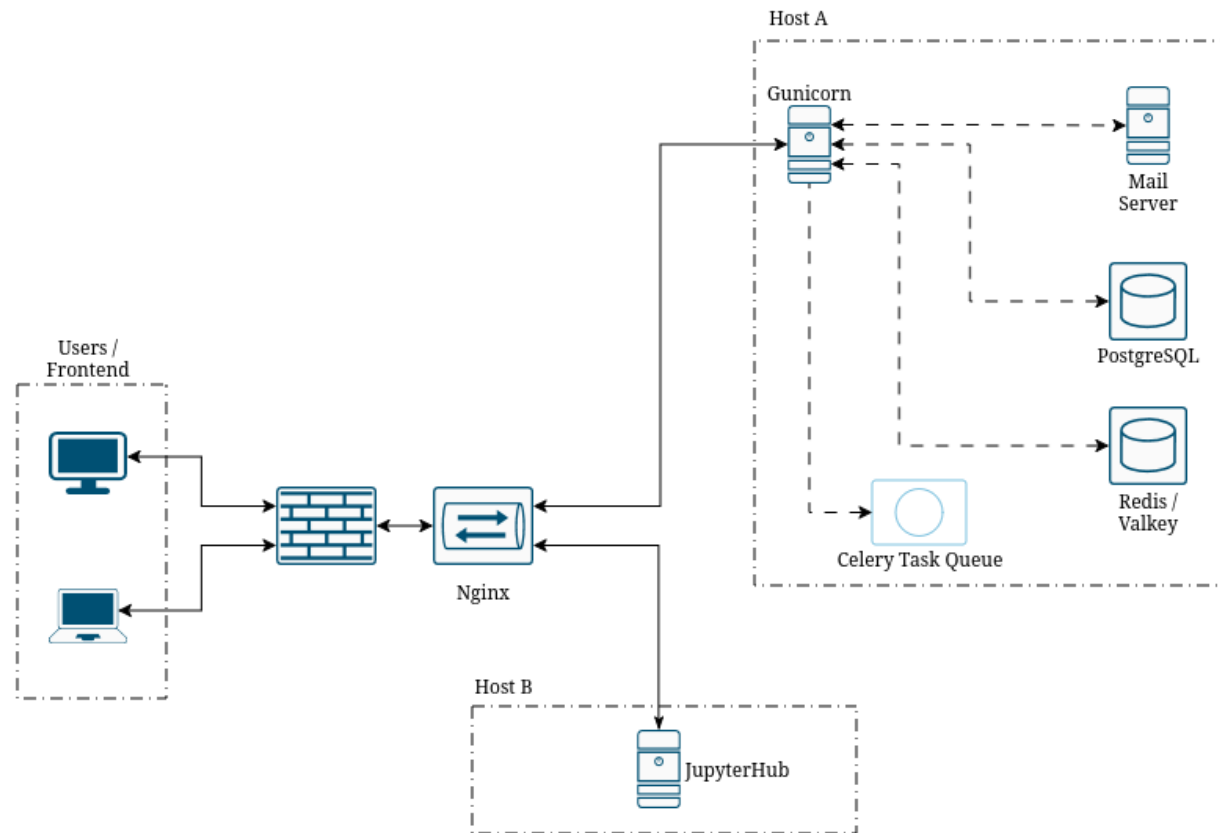
Arhitektura – Gateway 1 / 4

- Djeluje kao posrednik za upravljanje identitetima – pružatelj usluga i identiteta.
- Sprema i preslikava vanjske identitete u interne korisničke profile.
- Pružatelj identiteta za gostujuće korisnike koji nemaju institucijski IdP, podaci se čuvaju u PostgreSQL bazi.
- Pružatelj identiteta za JupyterHub - JupyterHub nikada nije izravno izložen vanjskim sustavima.
- Osigurava Single Sign-On tok za cijelu platformu.

Arhitektura – Gateway 2 / 4

- Životni ciklus sandbox okruženja:
 - 1) Korisnik klikne “Pokreni Sandbox” na sučelju
 - 2) Gateway šalje zahtjeve JupyterHub REST API-ju:
 - 1) Podešava JupyterHub za projekt – korisnici, grupe, dozvole za dijeljenje, itd.
 - 2) Pokreće instanciranje Jupyter poslužitelja.
 - 3) Dostavlja korisniku URL za ulazak na Jupyter Lab.
- Kontejneri / podovi se asinkroni brišu u pozadini - korisnik vidi trenutačno brisanje, ali dekomisioniranje teče bez blokiranja korisničke sesije
- Projekti su logičke grupacije, a ne instance po korisniku - svi suradnici konvergiraju na jedan kontejner / pod.

Arhitektura - Gateway 3 / 4



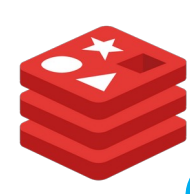
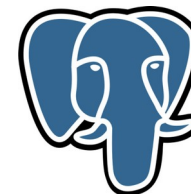
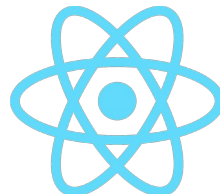
Arhitektura – Gateway 4 / 4

Frontend

- React SPA / Typescript
- Backend komunikacija kroz JSON API.
- JSON Web Token

Backend

- Django Rest Framework
- Nginx / Gunicorn
- PostgreSQL
- Redis / Valkey

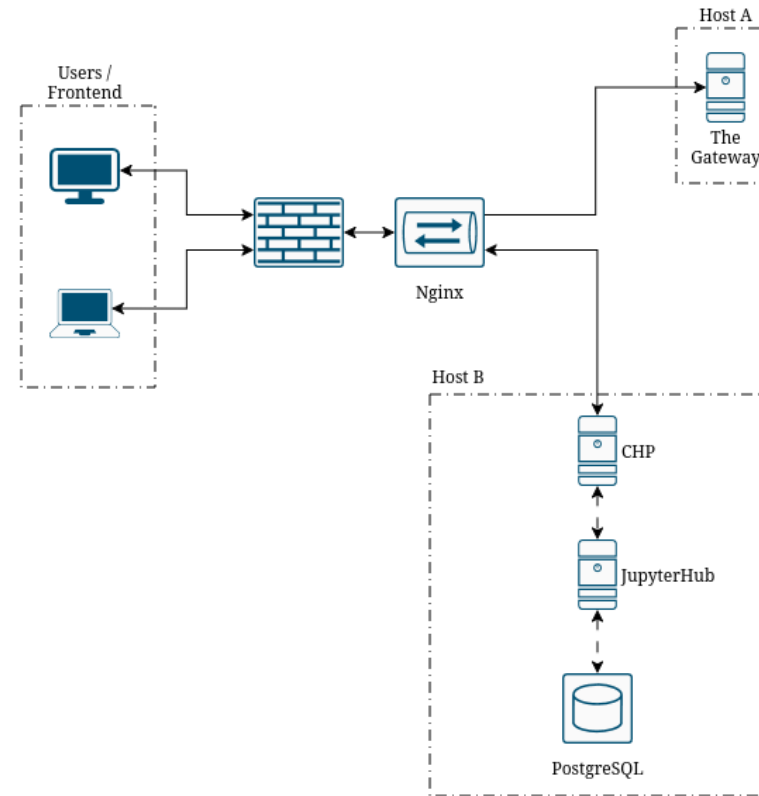


Arhitektura – JupyterHub 1 / 2

- Pruža REST API Gateway aplikaciji - Gateway je registriran kao administrativna usluga s povišenim privilegijama.
- Autentifikacija je delegirana natrag Gateway aplikaciji putem OIDC — JupyterHub inicira rukovanje radi provjere valjanosti sesije; nikada sam ne drži identitet.
- Usmjeravanje više korisnika:
 - Standardni JupyterHub svakom korisniku daje vlastiti izolirani poslužitelj
 - AI4Scale ovo mijenja: svi članovi projekta usmjeravaju se na isti poslužiteljski proces (isti kontejner / pod)
 - Ovaj model omogućuje uređivanje u stvarnom vremenu.

Arhitektura - JupyterHub 2 / 2

- Nginx - TLS terminacija
- Configurable HTTP Proxy (CHP) - dinamičko usmjeravanje korisničkih sesija; jedina javno izložena komponenta JupyterHuba.
- JupyterHub REST API - dostupan isključivo Gateway aplikaciji.
- PostgreSQL - korisničke evidencije, sesijski tokeni, tablice usmjeravanja proxyja.



Arhitektura – Jupyter Lab 1 / 2

- Korisničko okruženje za učenje i eksperimentaciju.
- Svako okruženje je Jupyter Server koji radi unutar Docker kontejnera ili Kubernetes poda.
- Povezan s identitetom projektnog korisnika kojeg je dodijelio Gateway.
- Sve projektne članove CHP usmjerava putem WebSocket veza na istu poslužiteljsku instancu.
- Dijeljeni proces, dijeljeno memorijsko područje, dijeljene računalne jezgre.
- Stanje bilježnice, izvršavanje ćelija i izmjene koda sinkroniziraju se u stvarnom vremenu u preglednicima svih suradnika.

Arhitektura – Jupyter Lab 2 / 2

- Potpuna CUDA integracija - Jupyter jezgre mogu izravno koristiti GPU resurse poslužitelja.
- Python ML knjižnice: NumPy, Pandas, PyTorch, TensorFlow i ostalo.
- JupyterLab proširenja za sinkronizaciju stanja više korisnika i kolaborativno uređivanje u stvarnom vremenu.
- Skup unaprijed učitanih bilježnica - svaki novi projekt počinje s istim pripremljenim obrazovnim sadržajem.

Hardware poslužitelja

- GPU
 - 2x NVIDIA PRO 6000 Blackwell WS 96 GB
- CPU
 - Threadripper Pro 9995WX (96 cores, 2.5 / 5.4 GHz)
- Radna memorija
 - 8x 128 GB DDR5 ECC 5600 MHz
 - Ukupni kapacitet 1024 GB
- Pohrana
 - 1 + 8 TB M.2 NVMe PCIe 5.0 TLC
 - 15.36 TB 2.5" U.2 NVMe PCIe 4.0 (Kioxia CD8-R)
 - Ukupni kapacitet: 39.72 TB

Što slijedi?

- Migracija platforme na namjenski hardver za stabilan i skalabilan rad.
- Unapređenje korisničkog iskustva na temelju povratnih informacija iz testiranja.
- Definiranje korisničkih uloga i pripadajućih prava pristupa (student, asistent, instruktor i sl.).
- Strukturiranje i usklađivanje sadržaja s obrazovnim ciljevima programa.
- Uvođenje Copilot agenta kao inteligentnog asistenta za podršku korisnicima unutar platforme.
- Integracija s ostalim platformama AI4GOV-X ekosustava.

Pitanja?

Dino Aljević

Fakultet Informatike i Digitalnih Tehnologija

Sveučilište u Rijeci

dino.aljevic@uniri.hr

Rijeka, 09.03.2026.

