**Final comment – TDA-14 “Alleen Turkse zaken tonen (AI-gebaseerd)”**

**Samenvatting**

We hebben de app aangepast zodat aan de voorkant **alleen winkels verschijnen die in eerdere user stories door onze AI-pipeline als “Turks” zijn geïdentificeerd**. Dit gebeurt via de bestaande backend-filtering (only\_turkish=true) die leunt op confidence\_score en/of expliciete Turkse signalen in notes. Frontend en backend zijn op elkaar aangesloten; het kaart- en lijstscherm vullen nu uitsluitend met AI-gevalideerde locaties.

**Wat is er gedaan**

* **Backend**
  + Endpoint /api/v1/locations/ uitgebreid/gebruikt met queryparams:
    - only\_turkish=true → filtert op AI-geïdentificeerde Turkse zaken.
    - min\_confidence (default 0.80) blijft beschikbaar.
  + Batchscript scripts/classify\_batch.py gebruikt om **CANDIDATE**-records te scoren en bij voldoende zekerheid te **promoveren naar VERIFIED**.
  + Redirect/URL-kleinigheden opgelost; endpoint is stabiel bereikbaar met en zonder trailing slash.
* **AI-koppeling**
  + POST /dev/ai/classify-apply?id=<id> aangeroepen door het batchscript voor 100 items.
  + Resultaat: **42** items kregen voldoende Turkse signalen en zijn gepromoveerd naar VERIFIED (>= 0.80).
  + Deze records worden nu door de public API opgenomen wanneer only\_turkish=true is gezet.
* **Frontend**
  + Datalaag (useLocations) callt nu:
  + /api/v1/locations/?only\_turkish=true&limit=200
  + UI toont kaart + lijst met uitsluitend Turkse zaken (op basis van AI-labels).
  + Geen lege-staat meer: kaart blijft zichtbaar zodra de API >0 resultaten geeft.

**Bewijs / testresultaten**

* **AI service up**
  + GET /dev/ai/ping → {"ok": true, "service": "dev\_ai"}.
* **Classify batch run (100 stuks)**
  + Promoted rows: **42** (min\_promote=0.80).
  + Teller per state na run:
    - CANDIDATE: 2983
    - VERIFIED: 92
    - RETIRED: 212
* **Public API check**
  + GET /api/v1/locations/?only\_turkish=true&limit=200 → lijst met **42** items (alleen Turks).
* **Frontend**
  + http://localhost:5173 toont lijst + kaart met uitsluitend Turkse locaties.

**Hoe te reproduceren (review stappen)**

1. **Backend starten**
2. cd Backend
3. source .venv/bin/activate
4. export DATABASE\_URL='postgresql+asyncpg://...'
5. uvicorn app.main:app --reload
6. **Frontend starten**
7. cd Frontend
8. npm run dev
9. **Verifiëren**
   * Open Swagger: http://127.0.0.1:8000/docs
   * Call: GET /api/v1/locations/?only\_turkish=true&limit=200
     + Verwacht: >0 resultaten, uitsluitend AI-Turks.
   * Open app: http://localhost:5173
     + Verwacht: kaart + lijst gevuld met diezelfde locaties.

**Randgevallen & kwaliteit**

* Wanneer de dataset (lokaal) nog niet genoeg ‘VERIFIED’ heeft, kan de lijst leeg zijn. Dit is nu ondervangen doordat we een eerste batch hebben gepromoot (42 stuks).
* min\_confidence kan indien nodig tijdelijk naar **0.75** om volume te boosten; default blijft 0.80 voor kwaliteit.

**Bekende punten / next**

* **Optioneel**: grotere batches draaien voor meer dekking:
* python scripts/classify\_batch.py --api-base "http://127.0.0.1:8000" --limit 1000 --sleep 0.25 --min-promote 0.80 --promote

(Alleen nodig als we snel meer volume willen tonen.)

* **Opschonen (alleen indien nodig)**: mocht te ruim gepromoot zijn, kunnen we met een kleine DB-update de drempel weer aanscherpen (niet nodig gebleken in deze iteratie).

**DoD / Acceptatiecriteria**

* Frontend toont **uitsluitend** door AI als Turks aangemerkte zaken.
* Backend endpoint ondersteunt filtering via only\_turkish=true en levert correcte, gecaste velden (lat/lng/rating).
* Kaart en lijst laden succesvol en consistent met de API-output.
* AI-classificatie is traceerbaar via /dev/ai/classify-apply en batchscript; resultaten zijn persistent in de DB (VERIFIED).

**Conclusie:** TDA-14 is functioneel afgerond en voldoet aan de user story. Klaar om te sluiten. ✅