






Interactive Soil Health Map (ISHM) — MVP

Agent-facing web app + REST API to view soil layers on a map and basic stats. Built with **Micronaut 4 (Java 21)**, serves a Leaflet UI and JSON endpoints. PostGIS/ML hooks come later.

Features (MVP)

-  **Map view** (Leaflet) with demo GeoJSON (bbox query).
 -  **Stats panel** (Chart.js) showing totals, pH distribution, crop mix.
 -  **REST APIs**
 - `GET /map/soil/bbox?minx&miny&maxx&maxy` → GeoJSON FeatureCollection (stub).
 - `GET /dashboard/summary` → summary metrics (stub).
 -  Static assets served from `/public`.
 -  Ready to swap to **Postgres + PostGIS** for real data.
-

Stack

- **Backend:** Micronaut 4, Java 21, Gradle
 - **Frontend:** Static HTML + Leaflet + Chart.js (served by Micronaut)
 - **Dev Config:** `application.properties` (+ CORS on for localhost)
-

Prerequisites

- JDK **21**
 - Gradle wrapper in repo (`./gradlew`)
 - (Optional) Docker for PostGIS later
-

Getting Started

1) Project layout

```
src/  
  main/  
    java/com/ishm/...  
    resources/  
      application.properties  
    public/  
      index.html
```

2) Run in dev

```
./gradlew run  
# open http://localhost:8080/
```

3) Build a fat JAR (local "deploy")

```
./gradlew clean build  
java -jar build/libs/*-all.jar  
# http://localhost:8080/
```

Configuration

src/main/resources/application.properties

```
micronaut.application.name=interactive-soil-health-map  
  
# Serve static HTML/JS/CSS from /public  
micronaut.router.static-resources.default.enabled=true  
micronaut.router.static-resources.default.mapping=/**  
micronaut.router.static-resources.default.paths=classpath:public  
  
# Dev only  
micronaut.server.cors.enabled=true  
micronaut.security.enabled=false
```

API (MVP)

GET /map/soil/bbox

Query params (defaults to India if omitted):

- minx, miny, maxx, maxy (EPSG:4326)

Response (GeoJSON FeatureCollection, stubbed):

```
{  
  "type": "FeatureCollection",  
  "features": [ { "type": "Feature", "properties": { "shc": "SHC123", "ph": 7.2 },  
    "geometry": { "type": "Polygon", "coordinates": [ ... ] } } ]  
}
```

GET /dashboard/summary

```
{
  "totals": { "soilPlots":1240, "farmers":910, "districts":42,
  "retestNeeded":180 },
  "phHistogram": [ {"range":"6.5-7.5", "count":540}, ... ],
  "cropCounts": [ {"name":"Paddy","count":520}, ... ]
}
```

Smoke test

```
curl "http://localhost:8080/map/soil/bbox"
curl "http://localhost:8080/dashboard/summary"
```

Roadmap (next)

- Swap stub bbox to **PostGIS**:
 - `CREATE EXTENSION postgis;`
 - Table `soil_plots`(geom geometry(MultiPolygon,4326), ph double precision, ...)
 - Query via `ST_MakeEnvelope` + `ST_AsGeoJSON`.
- Add `/soil/shc/{id}` → **recommendations**, **/feedback**.
- JWT auth for agents.
- Micrometer + Prometheus for metrics.

Troubleshooting

- **404 on /**
Ensure `application.properties` is in `src/main/resources/` and `index.html` in `src/main/resources/public/`. Static mapping props set as above.
- **Controllers not found**
Keep packages under `com.ishm` (same as `Application` base package).
- **JavaFX/Android later**
They'll consume the same REST endpoints. Keep CORS on in dev.

Scripts (handy)

```
# run
./gradlew run

# rebuild & run jar
./gradlew clean build && java -jar build/libs/*-all.jar
```

License

TBD (e.g., Apache-2.0).