

MALTESE GeoAI Interface

Prototype Development Summary

PoliRuralPlus Athens Code Camp | June 2026



The Problem

Before this work, MALTESE's AI tools were only accessible if you could:

Create a JACKDAW account

Login was required for every user individually

Run a Python server locally

Technical setup on your own machine every time

Manually connect the MCP server

Navigate settings, enter a URL, connect by hand — each visit



MALTESE

The Solution

Web Interface

Purpose-built, MALTESE-branded - no account or setup required

Server-side Auth

Invisible to users, automatic

Auto MCP Connect

Registers the data server silently at the start of each chat

Soil Type Function

Real ERA GeoJSON data - 383 polygons, 18 soil classifications

Moisture Function

EWA sensor readings

Funding Function

Structured entries with links and automatic deadline filtering

LIVE DEMO

The Interface

Authentication & MCP - Under the Hood

1 User opens the site

No login screen. Nothing to configure.

2 Server fetches a Bearer token

POST to poliruralplus.eu/o/token/ — OAuth2 client credentials

3 Token cached & auto-refreshed

Valid ~10 hours. Renewed before expiry.

4 MCP server registered

POST /mcp/connect at the start

5 User asks a question

All the above happened in the background

LIVE DEMO

Funding Opportunities

MCP Server - Before vs After

Tool	Before	After
Soil Type	3 hardcoded types, non-functional	383-polygon GeoJSON, 18 types, area-weighted %
Soil Moisture	Returned hardcoded 'yes' always	EWA sensor CSV, timestamped, VWC threshold (25%)
Funding	Plain text, no links, no deadline awareness	Structured entries, links, auto deadline filtering

LIVE DEMO

Mobile Responsiveness

Thank you

maltese-project.eu

PoliRuralPlus Athens Code Camp | June 2026

maltese-project.eu