

| ATTORI e OGGETTI |
|---|
| 1) UTENTE : compie le azioni e dà gli input (click, immagini, città) |
| 2) APP UI : interfaccia Streamlit (file main.py) |
| 3) GeoLoader (GPS) : fornisce i dati geografici all'inizio (selectbox) |
| 4) AIEngine : gestisce l'interazione con l'AI Gemini e il prompt |
| 5) Gemini API : esegue l'analisi dell'immagine e restituisce i risultati in formato JSON |
| NOTA! : dovremo mettere le linee di vita adeguate |

| MESSAGGI |
|--|
| 1) L'UI fa creare una lista di comuni al modulo GeoLoader con la funzione carica_dati_geografici() ricevendo tale lista come risposta |
| 2) L'utente seleziona la città, carica l'immagine e clicca su "Analizza rifiuto" |
| 3) L'App UI delega l'analisi all'AI Engine passando i parametri img, key, città |
| 4) L'AI Engine richiede il servizio esterno dell'API Gemini che fornisce un messaggio di analisi del rifiuto in formato JSON. Quindi l'AI Engine trasferisce tali risultati all'App UI |
| 5) L'UI gestisce ed elabora la logica dei colori in base ai dati ricevuti (auto-call) e mostra i risultati all'utente |
| 6) (OPZIONALE) Se il rifiuto è Speciale e va smistato in un'Isola Ecologica l'Utente può cliccare il bottone "Isola Ecologica" ed essere reindirizzato su Google Maps |

| LOOP |
|---|
| 1) Opt (Utente-APP UI): se l'Utente vuole la posizione dell'isola ecologica e clicca il pulsante riceve la risposta, altrimenti nulla (IF senza ELSE) |

