

- utente
  - 1) inserire città
  - 2) visualizzare intera chiamata
  - 3) vis. stat: che ↗
  - 4) cancella dati salvati
  - 5) scelta filtri
- [in più] → 6) uscita dal sistema (chiusura programma)

Scelta filtro è caso d'uso specializzato; vis. stat è generale.

- amministratore
  - 1) leggere città
  - 2) fare chiamata API
  - 3) Restituire tutto l'output (interno JSON)
  - 4) chiedere all'utente le statistiche
  - 5) Farle visualizzare
  - 6) Chiedere filtri
  - 7) Applicare filtri su stat.
  - 8) Salvare ogni ora info sul VENTO su una domenica dati
  - 9) Salvare la struttura dati su file
  - 10) Conversione JSON in JSONObject o JSONArray.

country code > sys.country (GB, JP, IT) ...

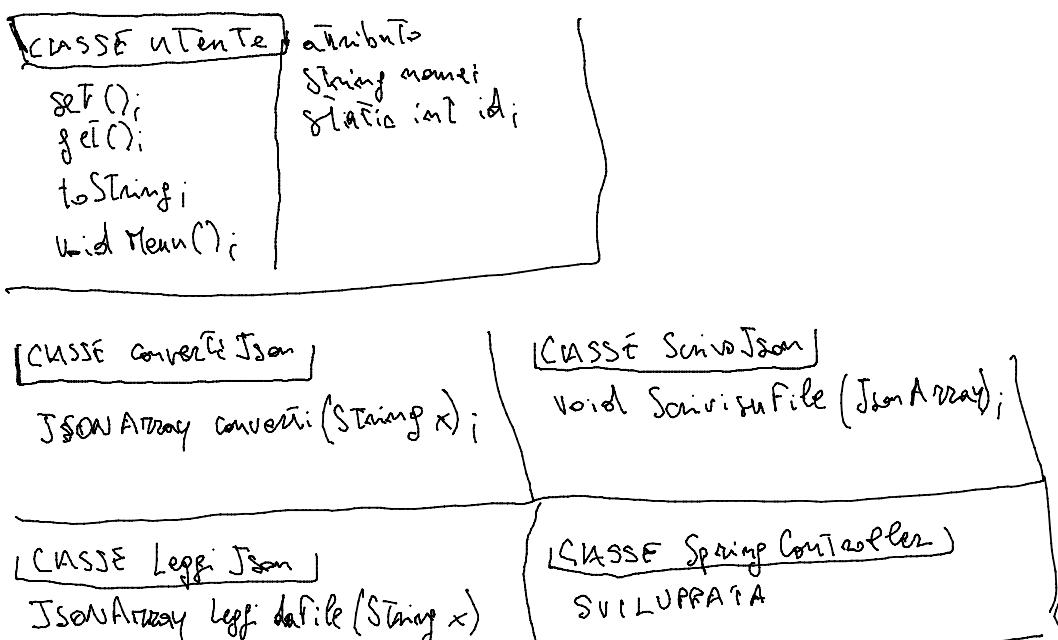
data lettura = dt.

Gli esempi di visualizza intera chiamate sono in più.

Amministratore (human) o system (non human)?

# applicaz. filtri fai, menù, settimanali (possono essere interface)

### DIAGRAMMA CLASSI



## CLASSE LeggiJson

JSONArray Leggi da file ('String x')

## CLASSE SpringController

Sviluppata

## CLASSE meteoController

Sviluppata

## CLASSE Vento

set();	float velocità;
get();	int direzione;
toString();	float raffica;

## INTERFACCIA APPLICAZIONE

Filter

## CLASSE STATISTICS

float velMedia (Vento x [])
float velMin (Vento x [])
float velMax (Vento x [])
float varianza (Vento x [])
float mediaDirzione (Vento x [])
float varianzaDirzione (Vento x [])