



# Linee guida progetti UF07:WEB

## Obiettivo dell'esame

Lo scopo dell'esame è valutare le conoscenze apprese durante le lezioni svolte in classe tramite la realizzazione di progetti.

## Valutazione

La valutazione sarà standard tramite un punteggio da 0 a 32, con la sufficienza fissata a 18.

Per raggiungere la **sufficienza** vanno **completati i progetti base 1 e 3**.

Il **progetto 2**, tutte le **parti avanzate** e la **documentazione** (*vedi sezione Documentazione sotto*) consentono di arrivare ai **32 punti massimi**.

## Consegna

I vari progetti dovranno essere creati utilizzando GitHub in modo da tenere traccia delle modifiche.

Per consegnare il progetto basterà allegare il **repository GitHub dove risiede il progetto**.

## Progetti

### 1) Task Manager

Creare una web app che permette all'utente di gestire una lista di attività. Per esempio:

- Obiettivi della giornata
- Processi di un computer

#### Progetto base

Il progetto deve poter gestire una **lista di oggetti** che rappresentano le vostre attività, permettendovi di **inserire**, e **rimuovere** le attività.

#### Progetto avanzato

Oltre all'inserimento e alla rimozione delle attività si gestisca:

- Modifica di un attività presente
  - Modificarne il nome
- Gestione degli stati di una singola attività:
  - Da fare
  - In corso
  - Completata
- In base al punto precedente poter filtrare la lista in base allo stato e mostrarla sulla pagina.
- Poter cercare un'attività specifica in base al nome.

### 2) Cronometro

Creare un cronometro digitale con pulsanti per avviare, fermare e resettare il conteggio del tempo.

## Progetto base

Creare un'interfaccia semplice con i pulsanti di `Start`, `Stop` e `Reset` e mostrare il tempo nel formato `MM:SS`.

## Progetto avanzato

- Implementare un pulsante `Giro` che salvi il tempo attuale in una lista.
  - Si mostri la lista con i rispettivi tempi dinamicamente nel formato `MM:SS:MS`
- Un'interfaccia con il tempo visualizzato in formato `MM:SS:MS`

## Note

Valutare l'utilizzo della funzione `setTimeout()` per la realizzazione del progetto.

## 3) Meteo App

Creare un'app che tramite una chiamata API recupera le informazioni meteo riguardo a una località.

## Progetto base

Dare latitudine e longitudine comporre la chiamata API per recuperare i dati meteo della località desiderata, salvarli localmente e mostrare una versione minimale delle informazioni.

L'URL API da usare è il seguente:

`https://api.open-meteo.com/v1/forecast?latitude=46.0679&longitude=11.1211&current=temperature_2m,relative_humidity_2m,precipitation,rain,cloud_cover,wind_`

```
https://api.open-meteo.com/v1/forecast?
latitude=46.0679&longitude=11.1211
&current=temperature_2m,relative_humidity_2m,precipitation,rain,cloud_cover,wind_speed_10m,weather_code
```

dove i parametri che ci interessano sono: `latitude` e `longitude`.

La chiamata API restituirà i seguenti dati:

```
{
  "latitude": 46.04,
  "longitude": 11.119999,
  "generationtime_ms": 0.06520748138427734,
  "utc_offset_seconds": 0,
  "timezone": "GMT",
  "timezone_abbreviation": "GMT",
  "elevation": 204.0,
  "current_units": {
    "time": "iso8601",
    "interval": "seconds",
    "temperature_2m": "°C",
    "relative_humidity_2m": "%",
    "precipitation": "mm",
    "rain": "mm",
    "cloud_cover": "%",
    "wind_speed_10m": "km/h",
    "weather_code": "wmo code"
  },
  "current": {
    "time": "2025-03-29T19:15",
    "interval": 900,
    "temperature_2m": 11.9,
    "relative_humidity_2m": 54,
    "precipitation": 0.00,
    "rain": 0.00,
```

```
"cloud_cover": 100,  
"wind_speed_10m": 9.8,  
"weather_code": 3  
}  
}
```

## Progetto avanzato


- Recuperare le coordinate locali del dispositivo tramite le API JS Navigation
- Documentare il significato del parametro `weather_code` presente nei dati.
  - Basandosi sul `weather_code` dopo le dovute ricerche mostrare delle icone relative al meteo.

## Note

Per la parte avanzata non abbiamo visto in classe cosa sono e come si usano le API Navigation. Lo scopo di questi ultimi punti è spronarvi a cercare e documentarvi su tecnologie e concetti che non avete mai visto o provato.

Allego la documentazione ufficiale per le API di Open-Meteo:

 Docs | Open-Meteo.com

Weather Forecast APIs with weather models from multiple national weather providers, combining the best models for accurate forecasts worldwide. Explore the API documentation to learn more about the available weather models, their origin countries, resolutions, forecast lengths, and update frequencies. Get detailed JSON hourly weather forecasts for up to 7 or 16 days by specifying the geographical coordinates and desired weather variables  
 <https://open-meteo.com/en/docs>

## Documentazione

La documentazione (*tecnica e non*) relativa ai singoli progetti è **facoltativa**.

La sua stesura

**comporta un aumento del punteggio** relativo al progetto e di conseguenza, al voto finale.

Le tipologie di documentazione richieste sono:

- Descrittiva
- Tecnica

### Documentazione descrittiva

Produrre un `README.md` che illustri il progetto descrivendone lo scopo, e le modalità di utilizzo.

### Documentazione tecnica

Si fa riferimento a tutta la documentazione relativa a codice, logiche e implementazioni effettuate per la realizzazione del progetto.

Questo tipo di documentazione è composta da commenti all'interno del codice e specifiche tecniche all'interno del file `README.md`.

Si consiglia di seguire la notazione JSDoc