

## CORSO DI PROGRAMMAZIONE JAVA AVANZATA

A.A. 2024/2025

PROF. LUCA GRECO

Si richiede di realizzare un'applicazione Java con interfaccia grafica (JavaFX ) per la gestione di un report di eventi sismici INGV ottenendo i dati da uno stream http.

L'applicazione deve avere un'interfaccia grafica conforme a quanto rappresentato in Figura 1, con finestra principale di dimensioni 800px X 600px.

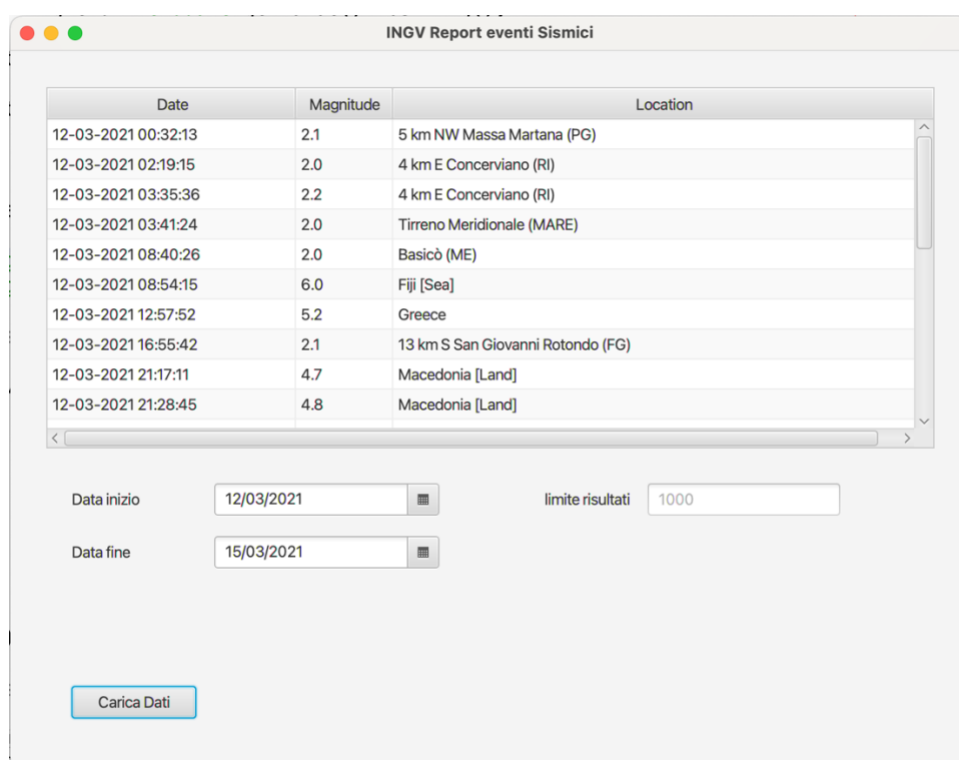


Figura 1 GUI del gestore

## OVERVIEW

L'applicazione dovrà poter importare il report tramite i servizi web disponibili all'URL:

<http://webservices.ingv.it/fdsnws/event/1/query?starttime=2020-11-18T00%3A00%3A00&endtime=2020-11-25T23%3A59%3A59&minmag=2&maxmag=10&mindepth=-10&maxdepth=1000&minlat=-90&maxlat=90&minlon=-180&maxlon=180&minversion=100&orderby=time-asc&format=text&limit=10000>

Cliccare sul link per verificare il formato dei dati.

I dati sono forniti dal servizio mediante uno stream testuale ed organizzati nella seguente forma:

#EventID|Time|Latitude|Longitude|Depth/Km|Author|Catalog|Contributor|ContributorID|MagType|Magnitude|MagAuthor|EventLocationName

L'applicazione dovrà consentire di **scegliere la data di inizio e di fine del report**, nonché indicare il numero **limite** dei risultati da visualizzare. A questo scopo sarà necessario modificare programmaticamente l'URL di richiesta del servizio.

Si richiede inoltre l'aggiunta di un campo di ricerca, dove è possibile filtrare le entry recuperate dal servizio.

## DETTAGLI SUL COMPORTAMENTO

All'avvio l'applicazione si presenta come in Figura 2. I datepicker che consentono di selezionare la data di inizio e di fine puntano alla data corrente, mentre il textfield associato al limite dei risultati ha un placeholder impostato al valore di default (il limite di default è 1000).

Cliccando sul pulsante *Carica Dati* si recuperano dallo stream di rete i contenuti da visualizzare nella tabella. Il recupero dei contenuti è gestito mediante un **Service**<sup>1</sup>, il cui **Task** effettua la lettura e il parsing dello stream di rete<sup>2</sup>. La tabella dovrà riportare per ogni evento i soli attributi come mostrato in Figura 1. L'avanzamento del processo di importazione del report è notificato all'utente da un *Progress Indicator*<sup>3</sup> (da posizionare vicino al bottone Carica Dati), che riporta la percentuale di eventi importati rispetto al limite di risultati visualizzabili.

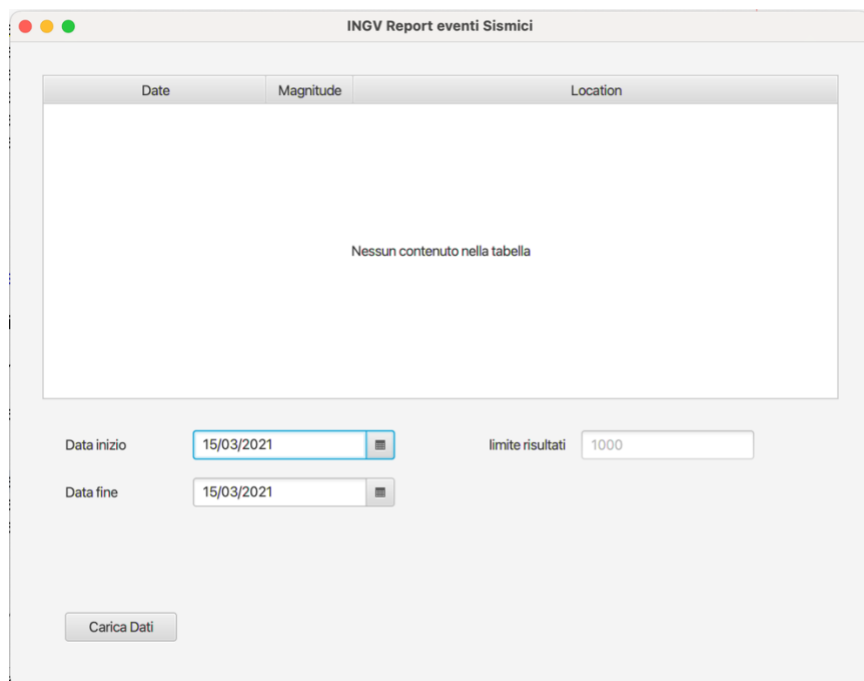


Figura 2 Stato iniziale

La tabella consente la selezione multipla delle righe. Una volta selezionata almeno una riga, tramite il menu contestuale<sup>4</sup>, (Figura 2) a cui si accede cliccando sul tasto destro, si abilita il pulsante *Salva selezione...* che consente di esportare su un file di testo CSV i soli contenuti selezionati riportando per ciascun evento solo gli attributi visualizzati.

Per ciascuna riga, il nuovo file dovrà avere il seguente formato:

**Date|Magnitude|Location**

<sup>1</sup> <https://docs.oracle.com/javase/8/javafx/api/javafx/concurrent/Service.html>

<sup>2</sup> <https://docs.oracle.com/javase/8/javafx/interoperability-tutorial/concurrency.htm>

<sup>3</sup> <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/ProgressIndicator.html>

<sup>4</sup> <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/ContextMenu.html>

L'utente dovrà poter scegliere il nome del file su cui salvare l'esportazione tramite un *File Chooser*<sup>5</sup>.

Si aggiunga infine una barra di ricerca<sup>6</sup> che consente di filtrare i dati acquisiti dal servizio (dopo il caricamento). La barra di ricerca deve interessare tutti i campi della tabella. Il campo va predisposto in alto a destra (sopra la tabella).

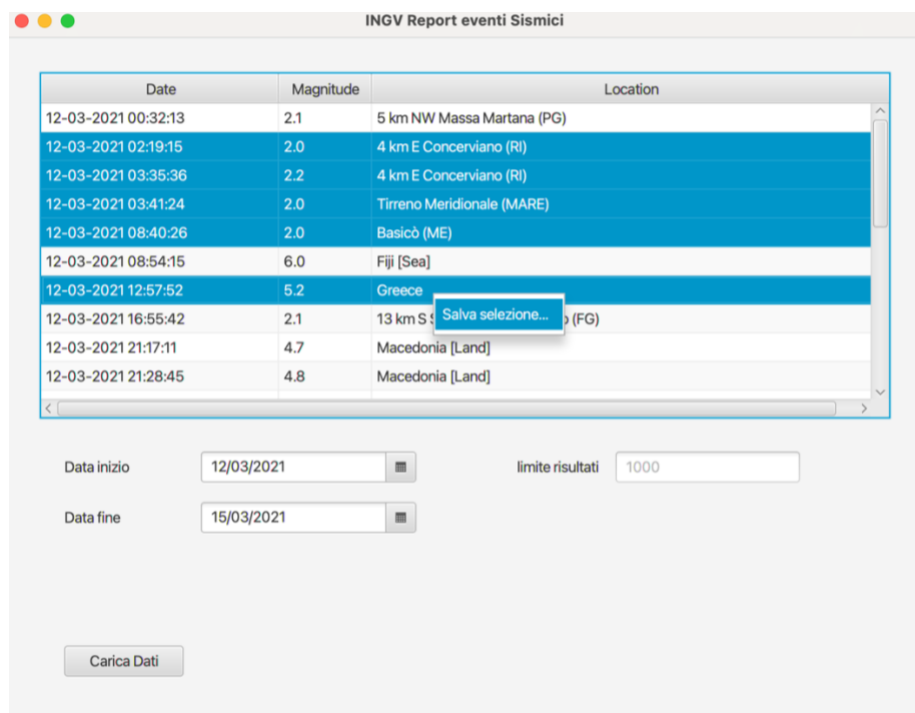


Figura 3 Selezione dei contenuti da esportare

## NOTE SULL'IMPLEMENTAZIONE

- Un Evento è rappresentato dalla classe **INGVEvent** che rappresenta l'evento e che include tutti gli attributi ad esso relativi. Scegliere per gli attributi il nome indicato sopra rispettando le convenzioni di Java per i nomi (per depth/km si può usare depthKm ) e il tipo di dato che si ritiene conveniente. Può essere utile implementare i getter e i setter per tutti gli attributi.
- Il **Service** responsabile del caricamento del report da stream di rete dovrà essere denominato **CaricaReportService** e avrà come attributi (almeno) l'URL remoto, il numero limite degli eventi da mostrare.
- In caso di implementazione JavaFX, l'applicazione va progettata seguendo il pattern MVC. La logica di funzionamento dell'applicazione è gestita da una classe *Controller* che implementa l'interfaccia *Initializable*. Il metodo astratto *initialize* dovrà essere implementato allo scopo di inizializzare tutti gli elementi utili alla creazione dell'interfaccia e per gestire eventuali operazioni preliminari necessarie per il corretto funzionamento. Il controller conterrà inoltre i metodi di gestione degli eventi associati ai componenti grafici per realizzare i comportamenti richiesti. Dovranno inoltre essere prodotti: il file FXML che contiene le specifiche della *View* e la classe principale dell'applicazione JavaFX da denominare *INGVReport*.

<sup>5</sup> [https://docs.oracle.com/javafx/2/ui\\_controls/file-chooser.htm](https://docs.oracle.com/javafx/2/ui_controls/file-chooser.htm)

<sup>6</sup> Si suggerisce l'uso del componente *FilteredList*