Politecnico
di Torino

Politecnico di Torino

Corso di Laurea in Ingegneria Gestionale
A.a. 2021/2022
Luglio 2022

Applicazione Java per il calcolo di un itinerario Interrail

Relatore: Fulvio Corno
Candidato: Filippo Castellarin

Contents

1 Proposta dello studente

1.1 Studente proponente

s270867 Castellarin Filippo

1.2 Descrizione del problema proposto

L'applicazione si propone di calcolare un itinerario percorribile in treno attraverso le stazioni delle città europee. Per fare ciò è necessario l'inserimento da parte dell'utente dei propri interessi turistici, inteso come le proprie preferenze in termini di attività da svolgere durante un viaggio, da quale città desidera partire, e il numero di giorni di viaggio del biglietto Interrail da lui acquistato.

1.2.1 Descrizione della rilevanza gestionale del problema

L'algoritmo tratta il calcolo di un itinerario turistico ottimale a partire dall'inserimento di semplici preferenze turistiche da parte del fruitore dell'applicazione. Per questo motivo, tale applicazione potrebbe risultare appropriata per tour operator e agenzie viaggi che si occupano nella definizione di itinerari turistici da vendere ai propri clienti.

1.3 Descrizione dei data-set

Le fonti dei dati provengono da:

- GitHub/Kaggle
- Eurostat

I database utilizzati sono:

- stations : contiene le informazioni circa la posizione geografica delle varie stazioni nel continente europeo. Questo data-set è distribuito con licenza ODC Open Database Licence; è possibile trovare la licenza completa al file `LICENCE.txt` del repository indicato.
- cities : mostra alcuni indicatori in termini di interessi culturali delle città europee che forniscono i dati all'Unione Europea, e di alcune città non facenti parte di essa.

Nota:
I database sono stati uniti tramite script `R` (file `tidyDatabase.Rmd` nella cartella Database) per permettere un uso più agevole del contenuto per il fine dell'applicazione in questione.
Il database completo è presente nel progetto alla cartella Database al nome di `cityTourStation.sql`.

1.4 Descrizione degli algoritmi coinvolti e istruzioni sull'uso

Avviato il programma, l'utente si trova di fronte a un'interfaccia grafica attraverso la quale può:

- selezionare lo stato di partenza, il che farà in modo che appena selezionato appaiano le città di quello stato;
- selezionare la città di partenza;
- selezionare il numero di giorni di viaggio;
- selezionare i propri interessi turistici (in base a quelli presenti nel database);
- premere il bottone 'Plan your trip' per avviare gli algoritmi per il calcolo del percorso;
- premere il bottone 'Reset' per azzerare le selezioni.

Ogni parametro è fondamentale per il calcolo dell'itinerario, pertanto nel caso in cui l'utente non selezioni qualche campo, l'applicazione segnalerà nell'area di testo sottostante la necessità di selezionare i campi mancanti.

L'algoritmo per il calcolo dell'itinerario sfrutta il principio della ricorsione a partire dalla città di origine selezionata dall'utente. In particolare, l'algoritmo va a massimizzare un parametro (dipeso dall'interesse turistico) calcolato per ogni città aggiunta alla soluzione parziale. L'algoritmo termina quando si è trovata la soluzione che va a massimizzare le città che risultano essere più affini per l'utente.
In caso di uguale peso, viene privilegiato l'itinerario con distanza minore.
Terminata l'elaborazione, il risultato viene mostrato in un'apposita area di testo.

NOTA:
Per poter rimanere il più verosimile all'effettiva quantità di chilometri percorribili durante un singolo giorno, il programma considera la possibilità di percorrere al più `1100km` (in linea d'aria) e di poter visitare più città nel caso esse abbiano una distanza pari o inferiore a `70km`.

2 Descrizione del problema

Il problema affronta la tematica dell'organizzazione di un itinerario turistico nei paesi facenti parte del territorio europeo.
Il lavoro del tour operator consiste nella creazione di itinerari turistici da vendere direttamente ai clienti finali, oppure nell'organizzare pacchetti turistici da vendere alle agenzie di viaggio che si occuperanno della distribuzione al dettaglio.

Per organizzare al meglio la propria attività lavorativa, al fine di essere sempre attivi e aggiornati in un settore in rapida evoluzione come quello del turismo, è necessario svolgere l'attività di base della ricerca e del calcolo di mete di viaggio suddivise in base agli interessi turistici dei possibili acquirenti nel più breve tempo possibile, per lasciare spazio al contatto diretto con le strutture ricettive dei luoghi individuati per creare un pacchetto turistico il più completo il possibile e che soddisfi al massimo le esigenze del cliente.

L'applicazione in questione si pone come obiettivo l'individuazione di tali mete turistiche, in base agli interessi turistico-culturali selezionati in input dall'utente.
Nello specifico, questa applicazione riguarda i percorsi di viaggio percorribili in treno, quindi adatti a viaggi di categoria ecologica (forte risparmio nelle emissioni di CO2 equivalente), sia a viaggi riguardanti il progetto Interrail (Eurail per extra-comunitari).
Per sua natura, l'applicazione si limita a mostrare delle possibili mete pertinenti a questa categoria di viaggio, senza fornire alcun contributo all'individuazione di eventuali strutture ricettive e tutto ciò che riguarda accordi per la fornitura di servizi nelle località turistiche.

3 Descrizione del data-set

Il data-set utilizzato proviene dall'unione di due database (vedi 1.3) riguardanti il primo la posizione geografica di tali città e se esse possiedono una stazione ferroviaria, e il secondo gli interessi turistici di varie città sparse all'interno del continente europeo.
L'unione dei due database è stata eseguita tramite uno script `R` che consiste nella lettura dei due file dati (xlsx e csv) provenienti dai database citati.
Dopodiché essi sono stati ripuliti da eventuali colonne di non interesse e modificati in modo da avere adeguata formattazione (es. il valore zero era codificato come ':').
Tutta la procedura è consultabile al file `tidyDatabase.nb.html` in formato html, oppure visualizzando il codice sorgente al file `tidyDatabase.Rmd`.
Il file frutto di tale elaborazione è denominato `stations_def.sql` e si trova nella cartella Database.

La pulizia di questi due data-set ha quindi contribuito alla creazione di uno nuovo contenente solamente i dati necessari all'elaborazione.
Nella figura 1 è mostrato il diagramma del database.

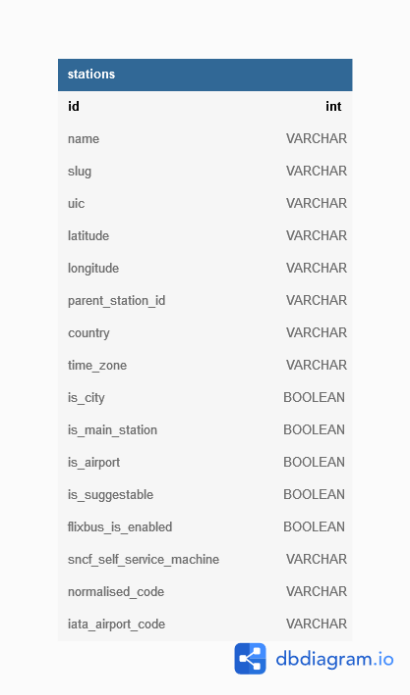

Figure 1: Diagramma DB

4 Descrizione degli algoritmi

Una volta avviato il programma, all'utente viene richiesto di selezionare i campi di input dagli appositi menù a tendina presenti sull'interfaccia utente. È bene notare, che ogni selezione è strettamente necessaria per elaborazione da parte del programma.

Una volta cliccato il pulsante `Plan your trip`, verranno avviati tutti gli algoritmi per la definizione dell'itinerario di viaggio.

Inizialmente, nella classe `FXMLController` vengono effettuate delle verifiche atte a determinare che l'utente abbia selezionato tutti i campi necessari per le operazioni, e se tale controllo non viene superato, nella casella di testo appare un messaggio contenente dei suggerimenti per poter usufruire delle funzionalità (es. Select origin city to continue.). Se tutti i controlli sono superati, seguendo il pattern Model View Controller (MVC), si delega alla classe `Model` la parte riguardante la logica applicativa. Tale classe richiamerà, quando necessario, la classe `InterrailDAO` per interrogare il database ed effettuare le operazioni necessarie di estrapolazione dati.
Dopo aver quindi premuto il pulsante `Plan your trip`, l'applicazione si collega alla classe `Model` che crea un grafo nel quale vengono inserite tutte le città contenute nel database: queste città vengono collegate tra di loro con archi solo se la distanza (calcolata tramite la libreria `LatLngTool` a partire dai dati di longitudine e latitudine) è inferiore al parametro `TOT` supposto pari a `1100km`. Tale parametro è modificabile dal programmatore, e tiene conto di alcune considerazioni riguardanti il numero di chilometri verosimilmente percorribili in un giorno di viaggio.
Successivamente alla creazione del grafo, il programma procede al calcolo tramite algoritmo ricorsivo delle migliori destinazioni a partire dalla città di partenza. In sostanza, l'algoritmo genera a partire dalla città di origine una lista di città prese dalla componente connessa (classe `ConnectivityInspector`), e da lì viene effettuata la procedura ricorsiva che mira a massimizzare il parametro di riferimento composto dalla somma della statistica relativa al parametro turistico-culturale selezionato dall'utente, prelevato in fase di creazione dell'oggetto `City` dal database.
Oltre alla massimizzazione del parametro indicato, l'algoritmo valuta anche la distanza totale percorsa, cercando di minimizzarla il più possibile.
Terminata questa fase, la funzione della classe `Model` che si è occupata di gestire il calcolo delle migliori destinazioni, passa il risultato alla classe `FXMLController` che fornirà in output all'utente le varie città di destinazione tramite apposita area di testo.

4.1 Diagramma delle classi principali

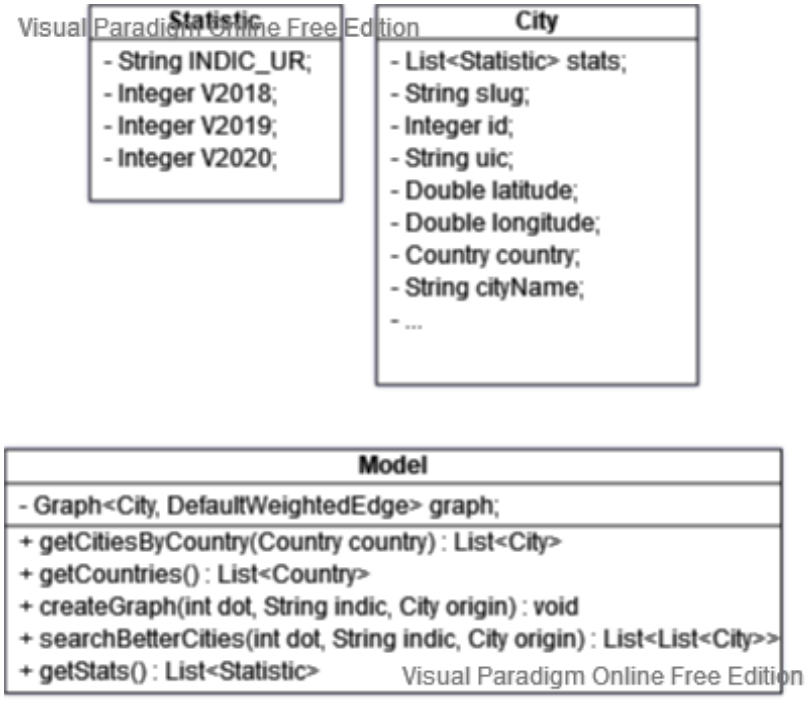

Figure 2: Diagramma delle classi principali

5 Risultati

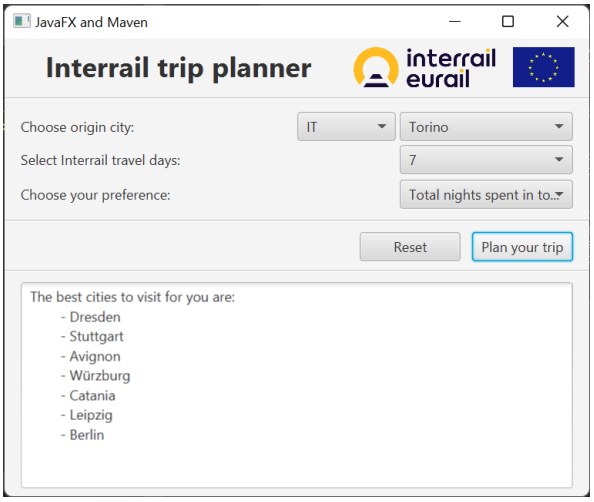

Figure 3: Torino trip

Nella cattura di figura 3, il programma è stato avviato selezionando come città di origine “Torino”, e da lì è stata avviata la ricerca del miglior percorso turistico da effettuare per interessi turistico-culturali orientati al numero di maggiori presenze turistiche.

Nella cattura di figura 4, il programma è stato avviato selezionando come città di origine “Murcia”, e da lì è stata avviata la ricerca del miglior percorso turistico da effettuare per interessi turistico-culturali orientati al numero di maggiori visite presso

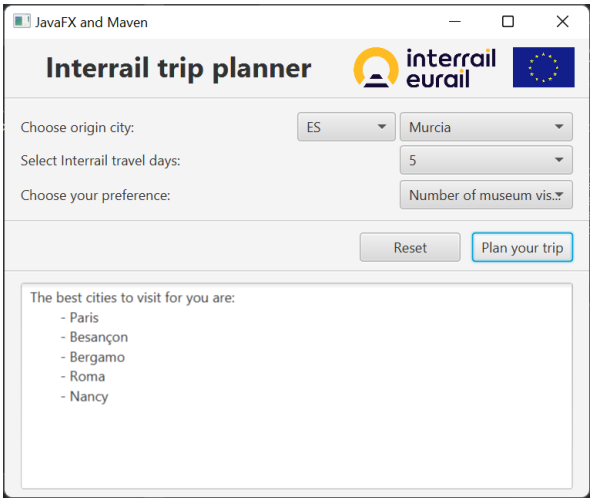

Figure 4: Murcia trip

i musei delle località.

Nella cattura di figura 5, il programma è stato avviato selezionando come città di

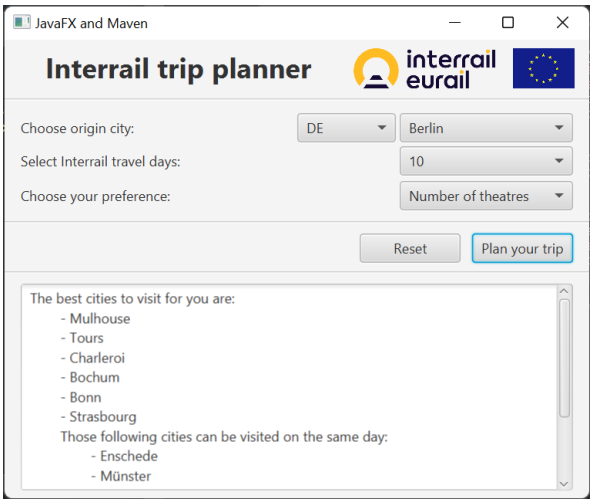

Figure 5: Berlin trip

origine "Berlino", e da lì è stata avviata la ricerca del miglior percorso turistico da effettuare per interessi turistico-culturali orientati al numero di opere teatrali.

5.1 Link al video dimostrativo

Video dimostrativo

5.2 Tabella dei risultati ottenuti

Stato	Città	Giorni	Interesse	Tempo (ms)
IT	Torino	7	Night stays	113
DE	Berlin	10	Theatres	3523
ES	Murcia	5	Night stays	18
FR	Paris	5	Theatres	15

Si può notare come al crescere del numero di giorni di viaggio (e quindi del numero di città fornite in output) l'algoritmo fatichi a trovare una soluzione in tempi ristretti. Fino a 7 fornisce soluzione in tempo istantaneo, mentre per valori superiori, l'attesa può anche durare pochi secondi.

6 Considerazioni finali

6.1 Punti di forza

L'applicazione è molto semplice e permette con estrema rapidità la valutazione di un itinerario turistico basandosi su semplici interessi turistico-culturali.
L'interfaccia è estremamente intuitiva, e ciò consente anche a utenti poco esperti di poter interfacciarsi con la stessa, senza che vi siano problemi di comprensione circa le funzionalità.
L'area di testo contenente il contenuto del risultato dell'elaborazione fornisce i risultati in maniera altresì semplice, in modo tale che l'utente possa vedere i nomi delle città immediatamente, senza essere distratto da ulteriori elementi visivi che potrebbero distorcere l'attenzione.
In caso di input mancante, all'utente viene fornita un'istruzione precisa su come procedere per poter immettere tutte le informazioni necessarie per l'avvio dell'elaborazione.

6.2 Limiti

I limiti principali dell'applicazione in questione riguardano sicuramente la parte dei dati: essi sono prelevati da due fonti diverse (e già questo di per sé genera problemi di unione dei dati). Inoltre, i dati che concernono il turismo e la cultura nel continente europeo sono poco precisi e vaghi.

L'applicazione presenta anche alcuni limiti della capacità computazionale, che per determinati indicatori e per un numero maggiore o uguale a 10 giorni di viaggio le tempistiche si allungano.

6.3 Conclusioni

L'applicazione così come si presenta si presta a un utilizzo per ambito non direttamente professionale, per via della sua semplicità e della poca accuratezza dei dati utilizzati.
L'applicazione si pone come obiettivo la velocizzazione della definizione di itinerari turistici da parte di tour operator e agenzie di viaggio che operino nel settore del turismo e in particolare nella definizione di pacchetti turistici da vendere al dettaglio.
Come risultati, essa fornisce i principali luoghi più affini agli interessi turistici selezionati (considerati i limiti di cui 6.2).
Un possibile miglioramento può essere apportato alla parte dei dati, poiché i dataset utilizzati non sono stati raccolti in modo completo e uniforme tra i vari Paesi. Inoltre, poiché i dati sul turismo variano quotidianamente, è altresì possibile fornire funzionalità che si occupino di collegamento ai database via rete e del prelievo e analisi dei dati aggiornati. Potendo quindi disporre di strumenti aziendali e dati raccolti da tutto il settore del turismo privato, sicuramente l'applicazione risulterebbe più efficace e completa potendo poi analizzare più fette del mercato turistico.

7 Licenza d'uso