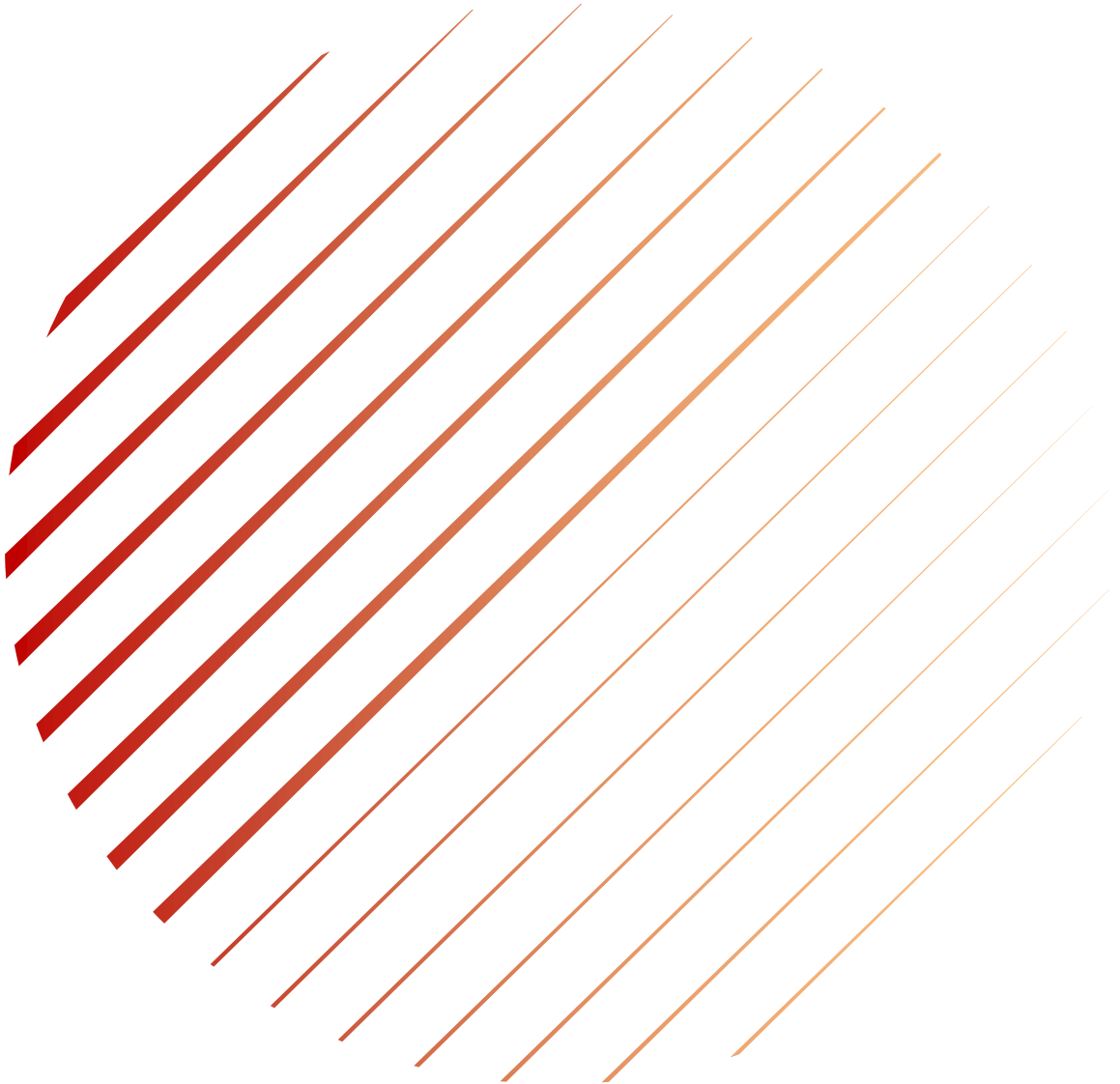


K-Means clustering

Caso di studio di Metodi Avanzati di Programmazione
AA 2022-2023



Realizzato Da

Simone De Girolamo 717450
s.degirolamo1@studenti.uniba.it

Ivan Digoia 716685
i.digoia3@studenti.uniba.it

SOMMARIO

1.	INTRODUZIONE.....	3
2.	INTRODUZIONE AL PROGETTO.....	5
3	DIAGRAMMI UML	6
4	GUIDA ALL' INSTALLAZIONE	15
5.	GUIDA UTENTE	16

1. INTRODUZIONE

1.1 L'algoritmo k-Means

L'algoritmo del k-Means clustering è un algoritmo di creazione di clustering utilizzato per suddividere i dati in gruppi omogenei basati sulla loro somiglianza. La prima idea del k-Means risale al 1956, l'anno successivo nel '57 il suo algoritmo standard verrà proposto da Stuart Lloyd, nel '65 Edward Forgy pubblicò lo stesso metodo e l'algoritmo fino al '67 prenderà il nome di Lloyd-Forgy anno in cui prenderà il nome di k-Means.

Esso è progettato per operare su dati omogenei. Ogni transazione è vista come un insieme di elementi. Dato un numero di cluster K , l'algoritmo divide l'insieme totale delle transazioni in K gruppi (Cluster) generati in base alla distanza tra ogni elemento, e ne calcola un Centroide, ovvero un punto medio tra tutte le transazioni di quel cluster.

K-Means utilizza un approccio “dal basso verso l'alto”, in cui i sottoinsiemi vengono creati casualmente dai dati. Ad ogni iterazione assegna ciascuna osservazione al centroide più vicino e ricalcola i centroidi. L'algoritmo termina quando raggiunge il numero massimo di iterazioni.

K-Means utilizza una struttura di partizionamento per suddividere una grande mole di dati in cluster gestibili. Genera K celle, dove ogni cella rappresenta un cluster, e ogni transazione viene assegnata al cluster più vicino. Quindi procede a minimizzare la varianza all'interno di ogni cluster calcolando la distanza tra le singole transazioni ed il centroide. Infine, ripete questo processo spostando gli elementi in base a quanto sono distanti dal prototipo del centroide calcolato.

1.2 Limiti

L'algoritmo K-Means soffre di una serie di inefficienze. È necessario scegliere manualmente il numero di cluster, e questo può influenzare significativamente i risultati. Infatti, un numero di cluster troppo piccolo rispetto al numero di transazioni potrebbe rendere i cluster stessi troppo grandi ed includere dati diversi, il che porta ad una sovrapposizione tra cluster. Viceversa, un grande numero di cluster porta gli stessi ad essere troppo specifici e potrebbero includere all'interno rumore o variazioni casuali nei dati

L'algoritmo inizia selezionando casualmente i centroidi a partire dai dati. Se i centroidi sono scelti in maniera subottimale, l'algoritmo potrebbe convergere verso un ottimo locale invece di un ottimo globale. Di conseguenza, è necessario eseguire l'algoritmo più volte con diverse inizializzazioni, così da poter selezionare il risultato migliore

2.INTRODUZIONE AL PROGETTO

2.1 Descrizione del progetto

Il software realizzato utilizza l'algoritmo k-Means, descritto nella sezione precedente, esso elabora dati da una tabella presente in un database di tipo MySQL.

Il progetto, risultato di esercitazioni, consiste in un'applicazione di tipo Client/Server.

Il server si occupa di ricevere le richieste di un client, il quale può effettuare le seguenti operazioni:

- Generare un numero di cluster partendo dai dati del database e li memorizza in un file .ser
- Caricare da un file .ser i cluster memorizzati

In entrambi i casi, il client dovrà specificare nei criteri di ricerca:

- Il nome della tabella da cui estrarre i dati dal database
- Il numero di cluster

Il server mostra inoltre informazioni sul client connesso:
le operazioni da esso richieste e il loro esito.

L'interfaccia grafica è stata sviluppata usando la tecnologia JavaFX; inoltre, è stato utilizzato SceneBuilder per la creazione dell'interfaccia grafica e CSS.

Nel progetto sono presenti entrambe le versioni, sia quella fruibile attraverso console, sia quella utilizzabile con l'interfaccia grafica. Nel presente documento verrà trattata solo l'estensione.

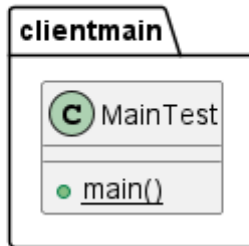
Nella sezione 3 sono riportati anche i diagrammi UML per il client e per il server. Inoltre, nella cartella "Javadoc" è stata allegata la Javadoc creata direttamente dall'IDE di sviluppo (IntelliJ). Nella sezione 5 del documento sono riportati esempi di esecuzione.

3 DIAGRAMMI UML

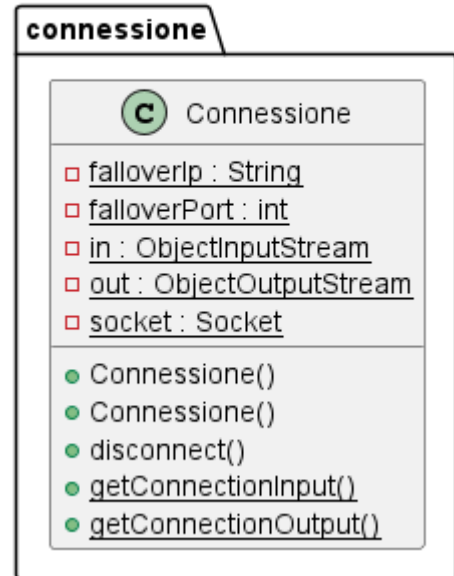
Segue la realizzazione dei diagrammi per la versione Base del MeansServer e KMeansClient

3.1 Client UML

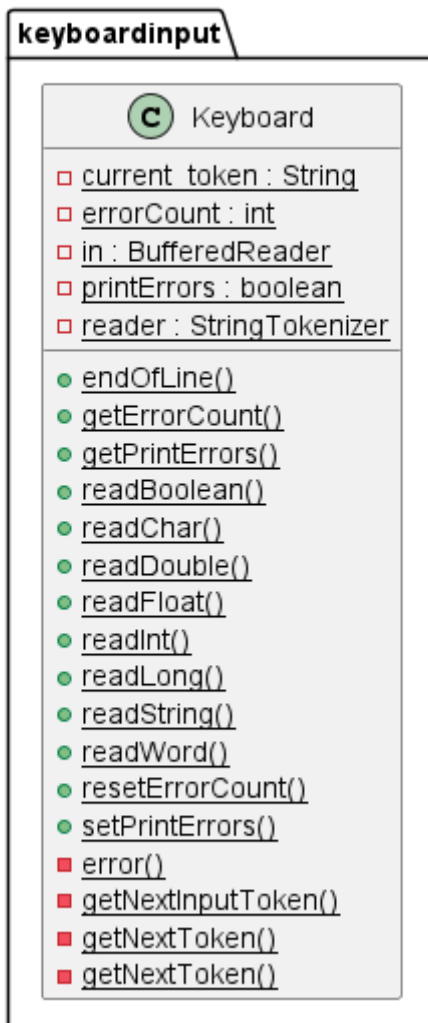
CLIENTMAIN's Class Diagram



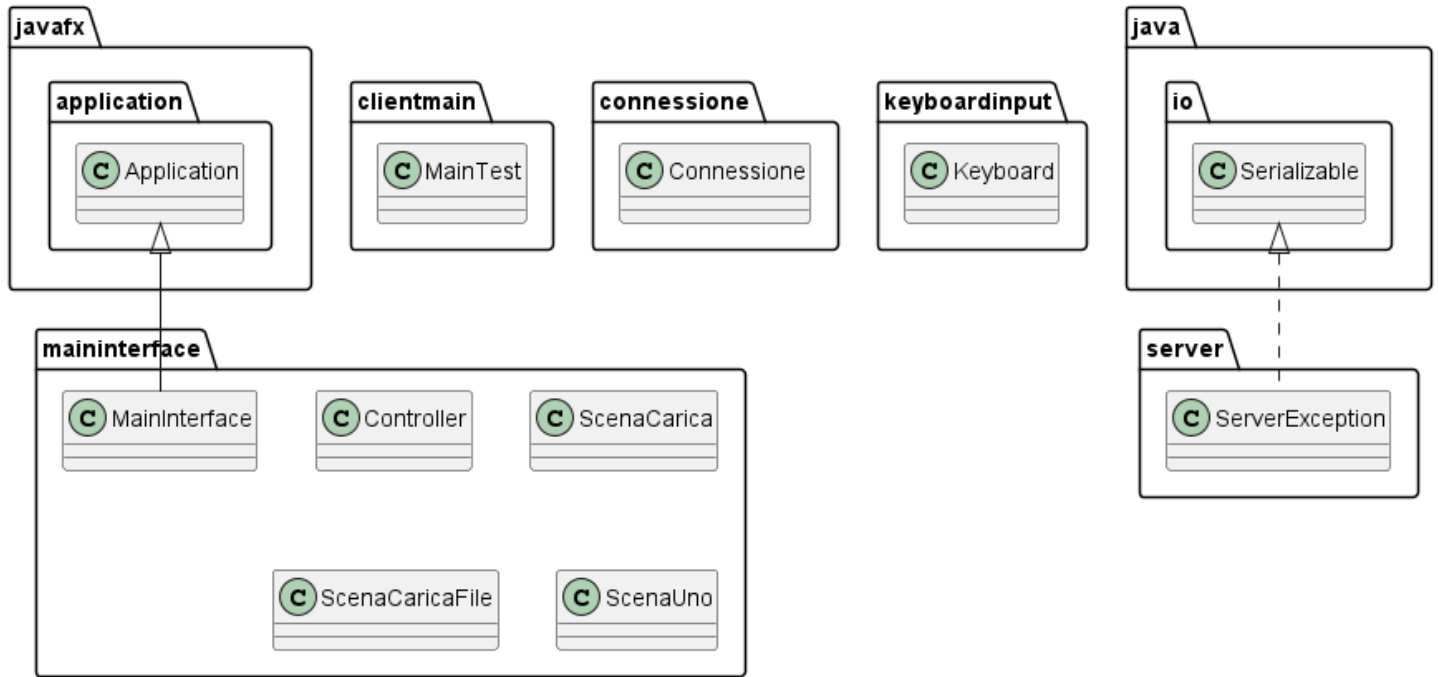
CONNESSIONE's Class Diagram



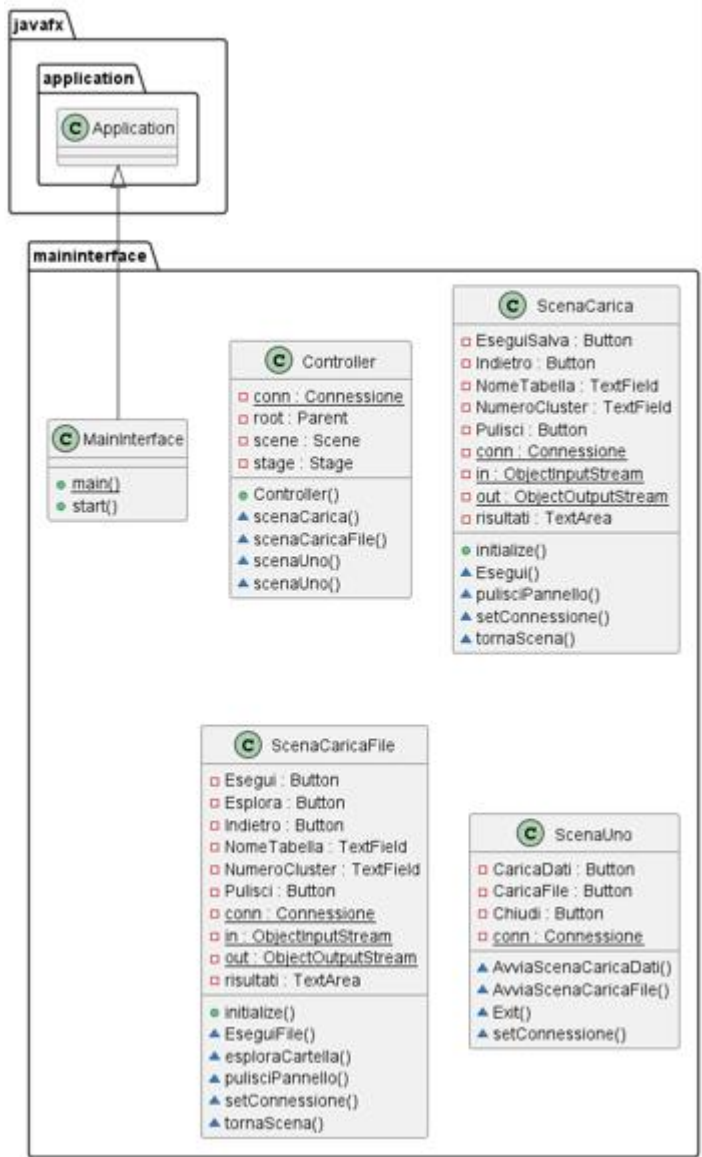
KEYBOARDINPUT's Class Diagram

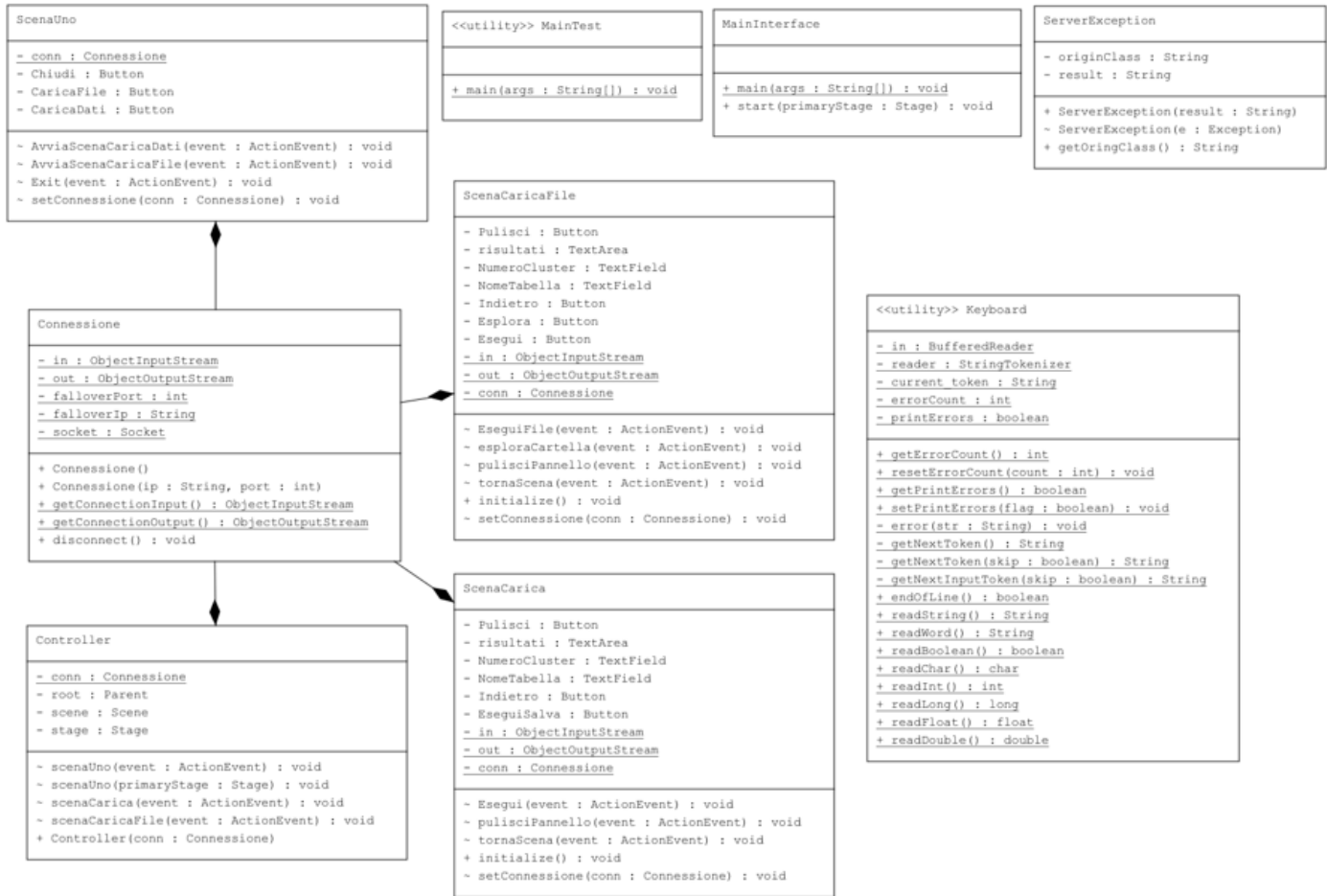


KMEANSCLIENT's Class Diagram



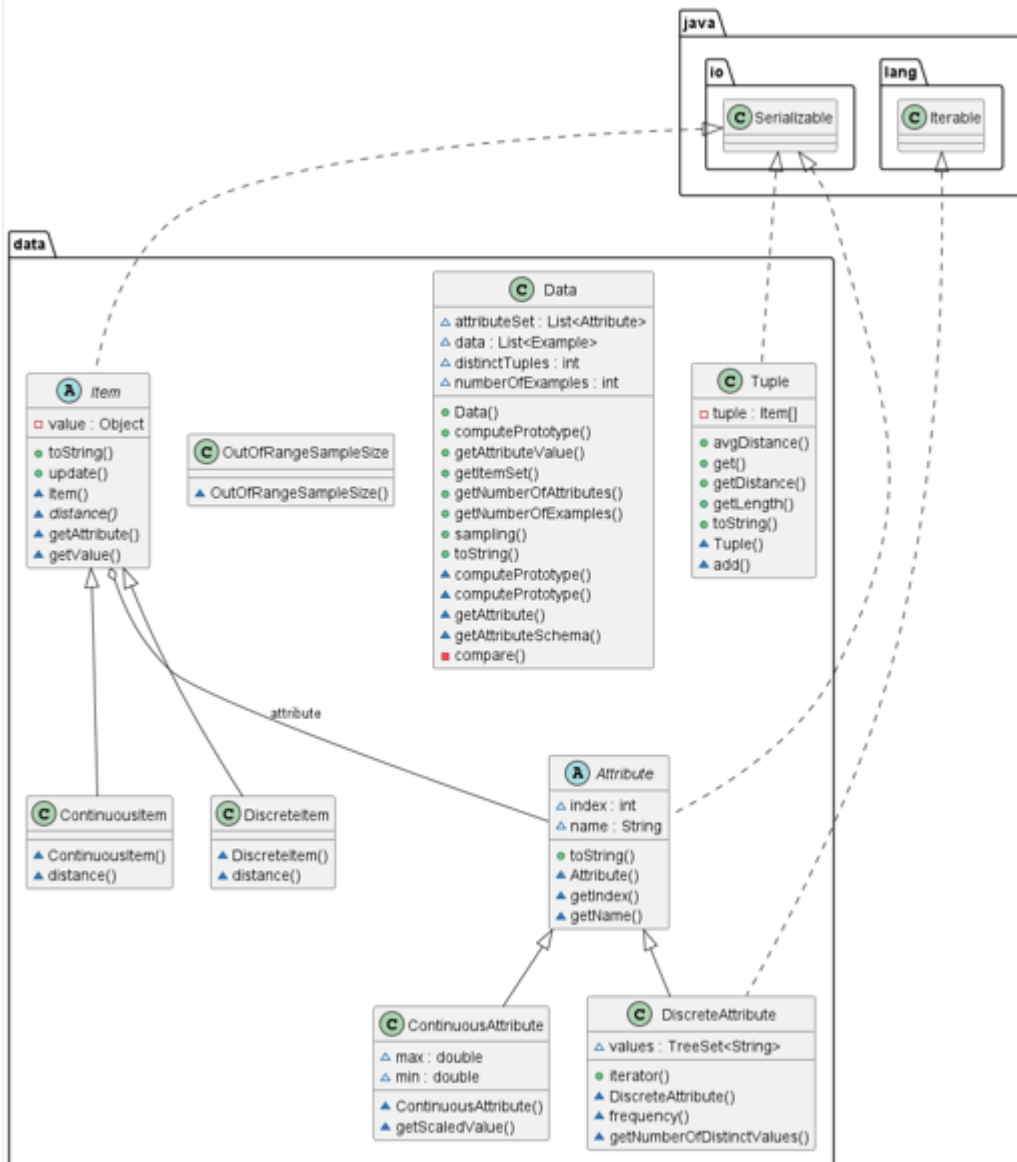
MAININTERFACE's Class Diagram



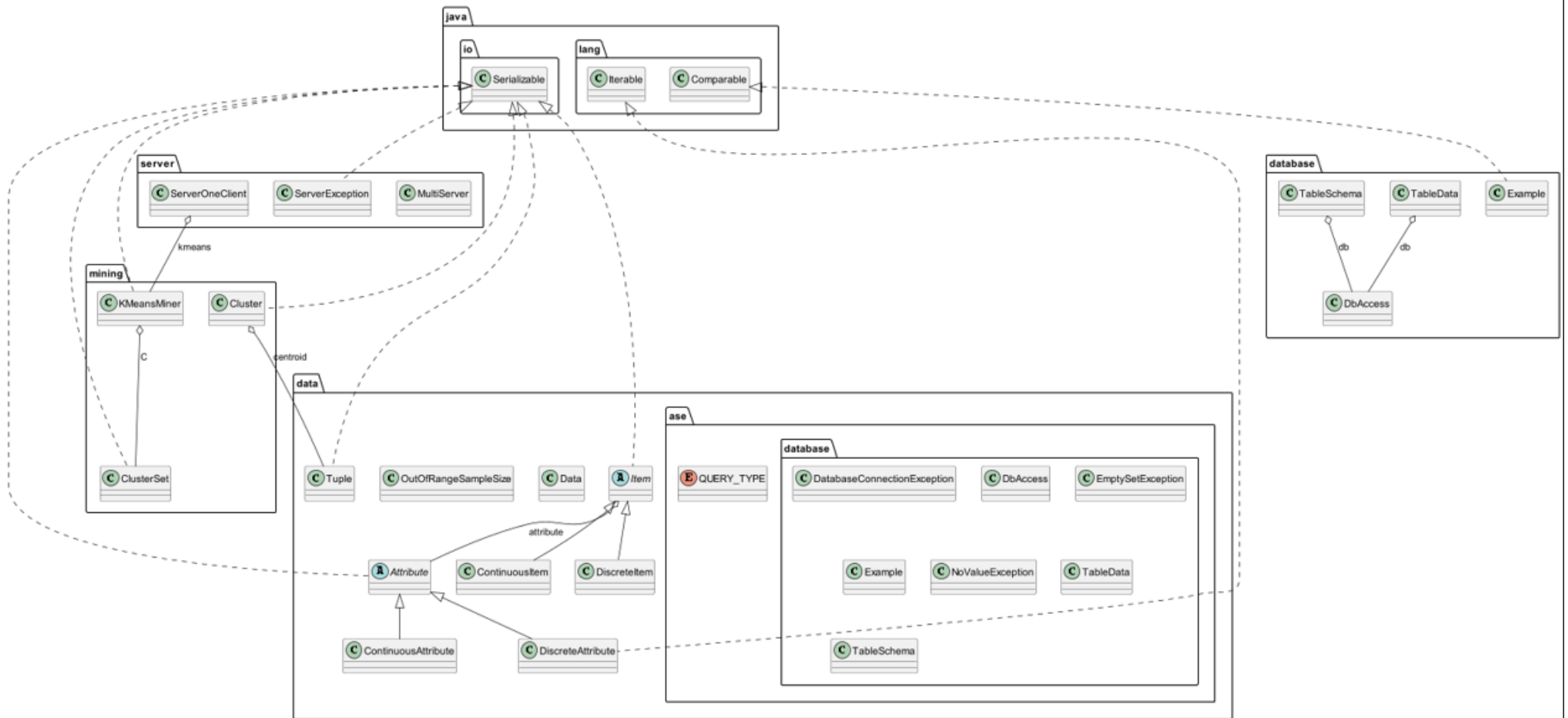


2.2 Server UML

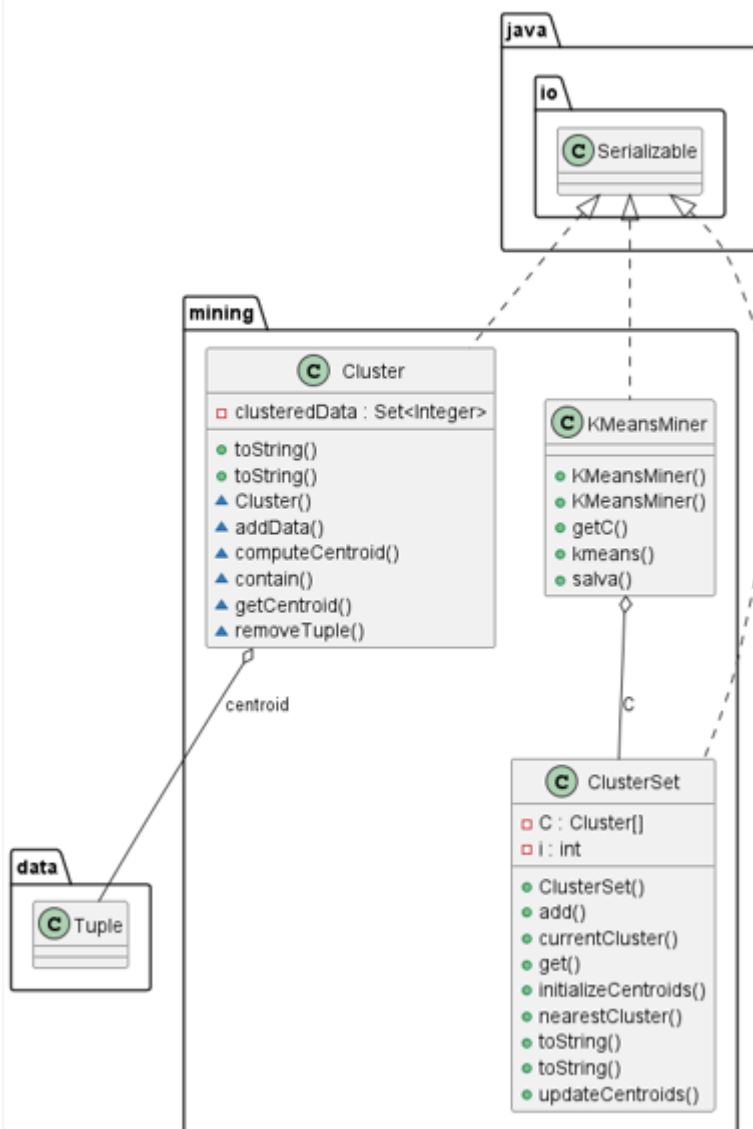
DATA'S Class Diagram

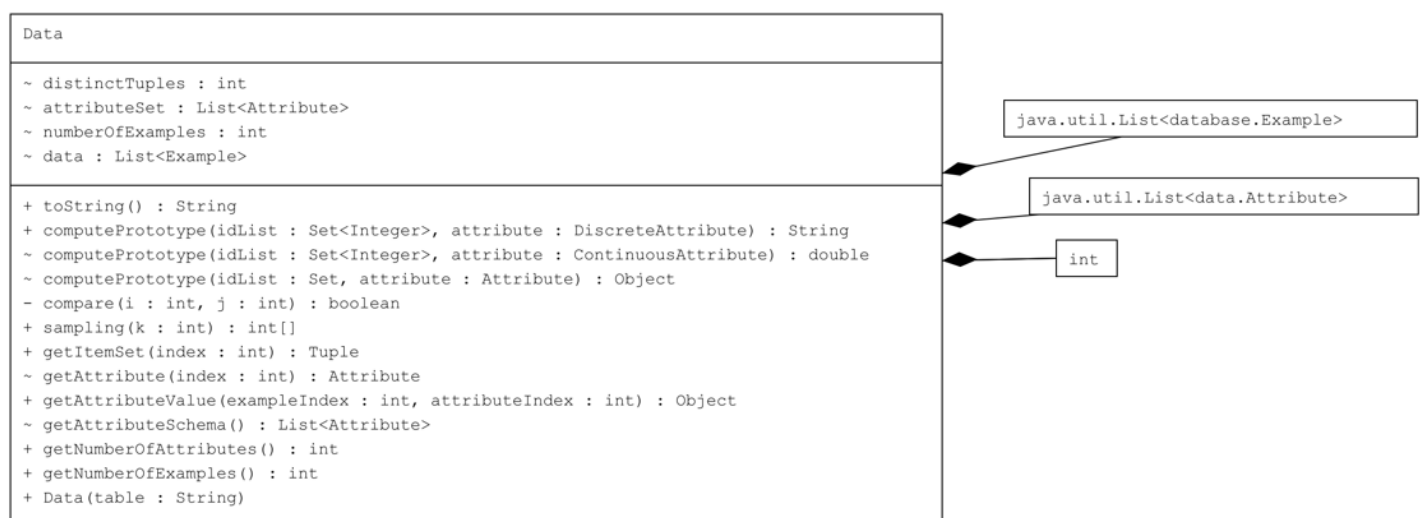
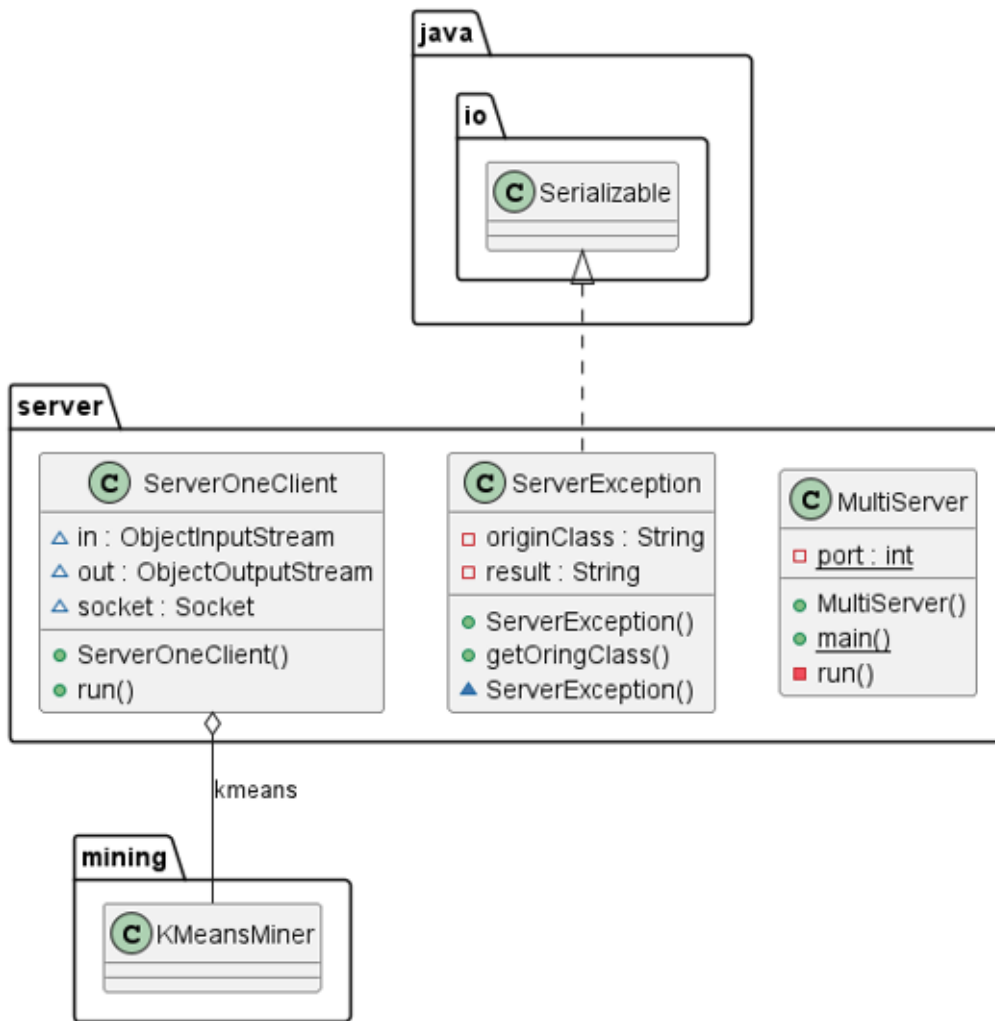


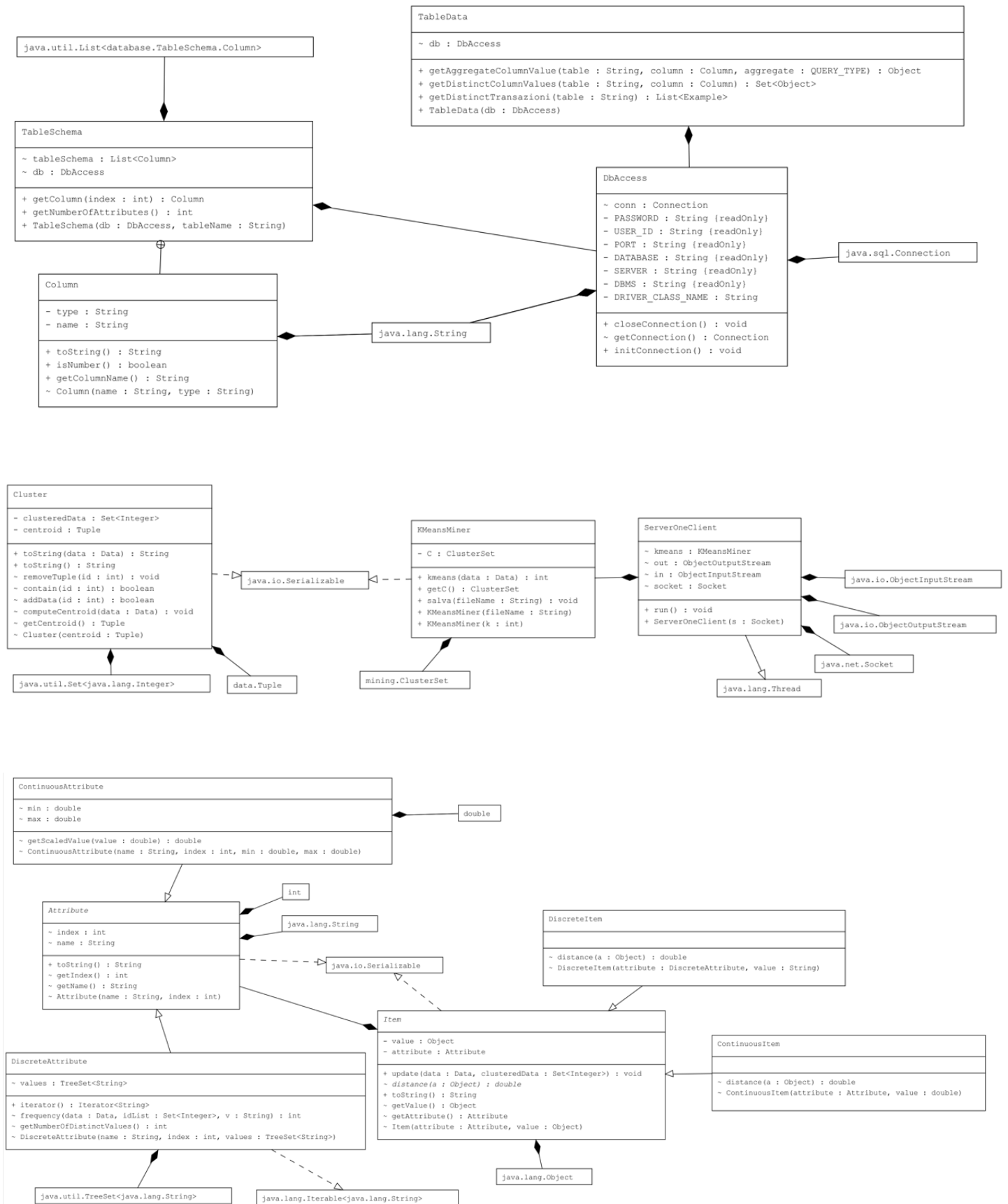
KMEANSSERVER's Class Diagram

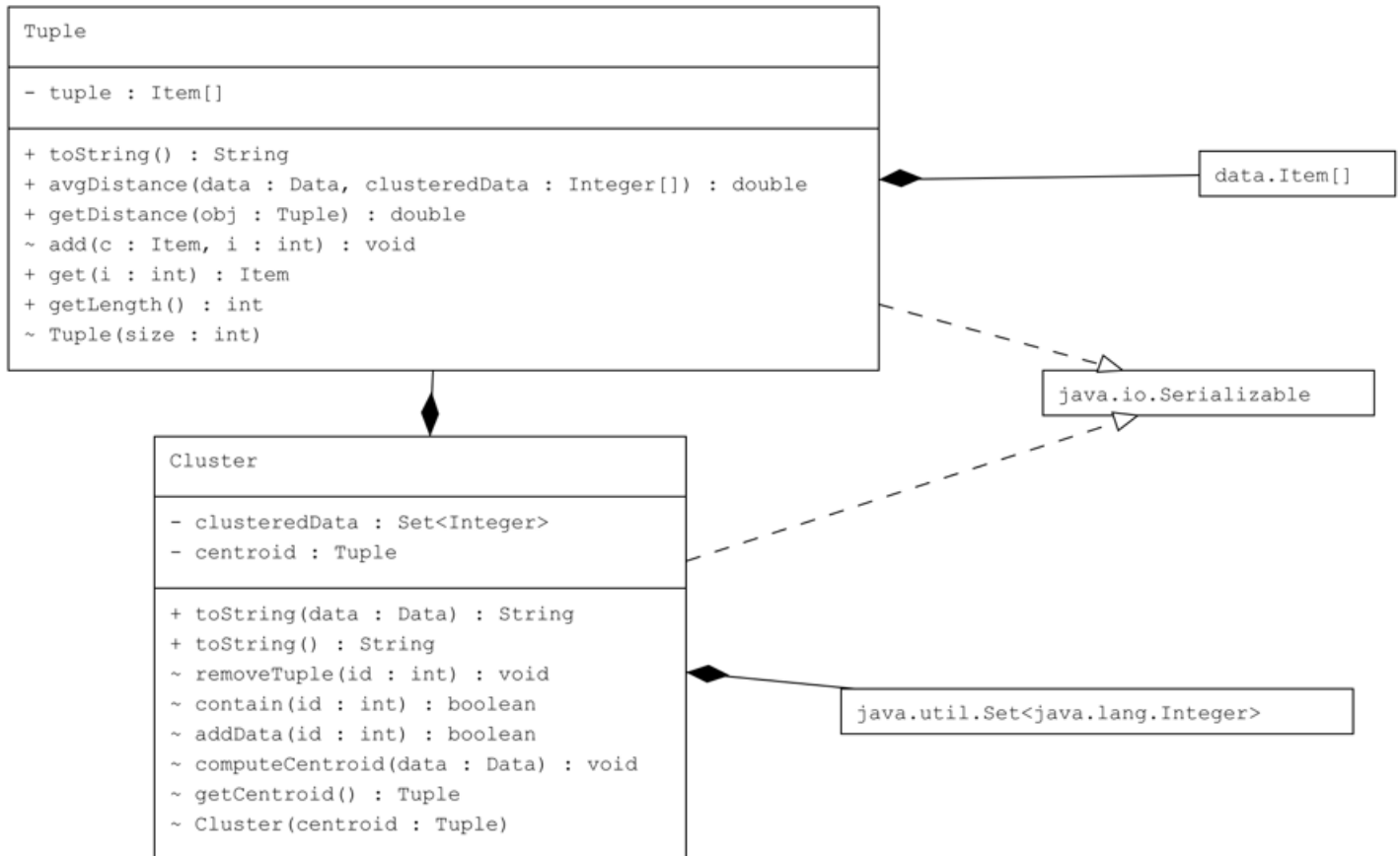


MINING's Class Diagram



SERVER's Class Diagram





4 GUIDA ALL' INSTALLAZIONE

4.1 Installazione Server

Per il corretto funzionamento del progetto lato server è necessario:

- Spostare l'intera cartella del progetto sul desktop;
- Installare MySQL 8.0;
- Installare Java Runtime Environment (JRE) versione 20;
- Avviare il server MySQL;
- Eseguire lo script MySQL presente nella cartella "SQL Connector". Tale script inizializza il database con tabelle e tuple di esempio.ⁱ

Per avviare il client è possibile aprire il file *Eseguibile Client.bat* contenuto nella cartella "Eseguibile/Estesa". Alternativamente, è possibile avviare il client tramite riga di comando indicando (partendo dalla cartella in cui si trova il file *Eseguibile/Estesa/KmeansClient.jar*):

- La directory in cui è contenuto il java.exe (se non è contenuto nel PATH)
- Il comando -jar che indica di avviare un file .jar

La riga sarà simile a:

```
C:\$PathTo$\java.exe -jar KMeansServer.jar
```

4.2 Installazione Client

Per il corretto funzionamento del progetto lato client è necessario:

- Installare Java Runtime Environment (JRE) versione 20;
- Avviare il server.ⁱⁱⁱⁱ

Per avviare il client è possibile aprire il file *Eseguibile Client.bat* contenuto nella cartella "Eseguibile/Estesa". Alternativamente, è possibile avviare il client tramite riga di comando indicando (partendo dalla cartella in cui si trova il file *Eseguibile/Estesa/KmeansClient.jar*):

- La directory in cui è contenuto il java.exe (se non è contenuto nel PATH);
- Il comando -jar che indica di avviare un file .jar;
- L'indirizzo IP a cui è collegato il server (di default 127.0.0.1);
- La porta su cui è in ascolto il server (di default 8080);
- Le librerie javafx necessarie all'avvio del programma (base, controls, graphics, media, fxml);

La riga sarà simile a:

```
C:\$PathTo$\java.exe -jar "./lib" --add-modules
```

```
javafx.base,javafx.controls,javafx.graphics,javafx.media,javafx.fxml -jar KmeansClient.jar 127.0.0.1 8080
```

5. GUIDA UTENTE

Nella cartella principale del progetto è presente una sottocartella “*File memorizzati*”, nella quale verranno salvati (e caricati) in file .ser tutti i pattern trovati. In essa sono presenti già dei file a scopo di esempio

Le tabelle di esempio presenti nello script MySQL si chiamano “*playtennis*”.

4.3 Guida all'interfaccia grafica

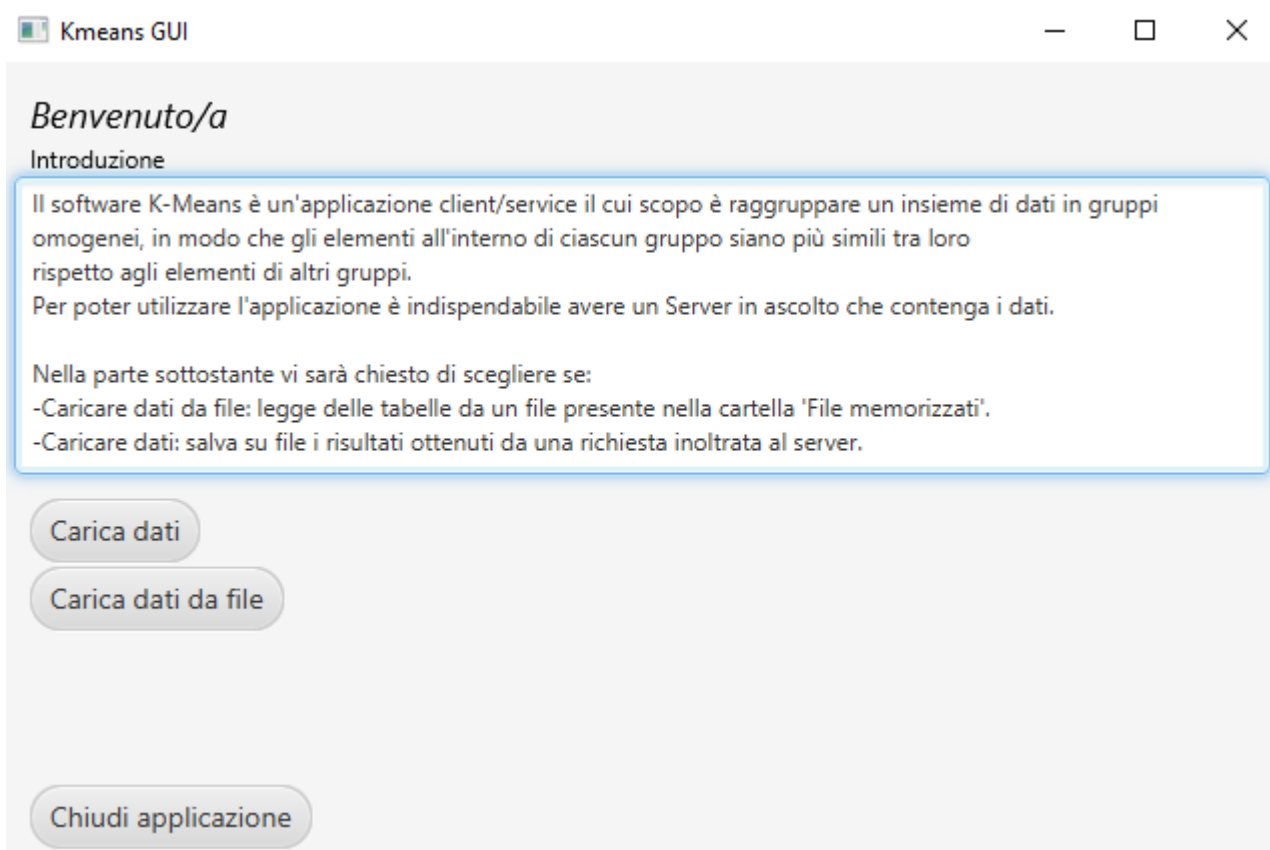
Nella cartella “*Eseguibile*” eseguire il file “*Eseguibile generale Esteso.bat*”. Si aprirà una schermata a linea di comando essa sarà il server che si avvia e successivamente si aprirà la prima schermata dell'interfaccia.

1) Avvio server:

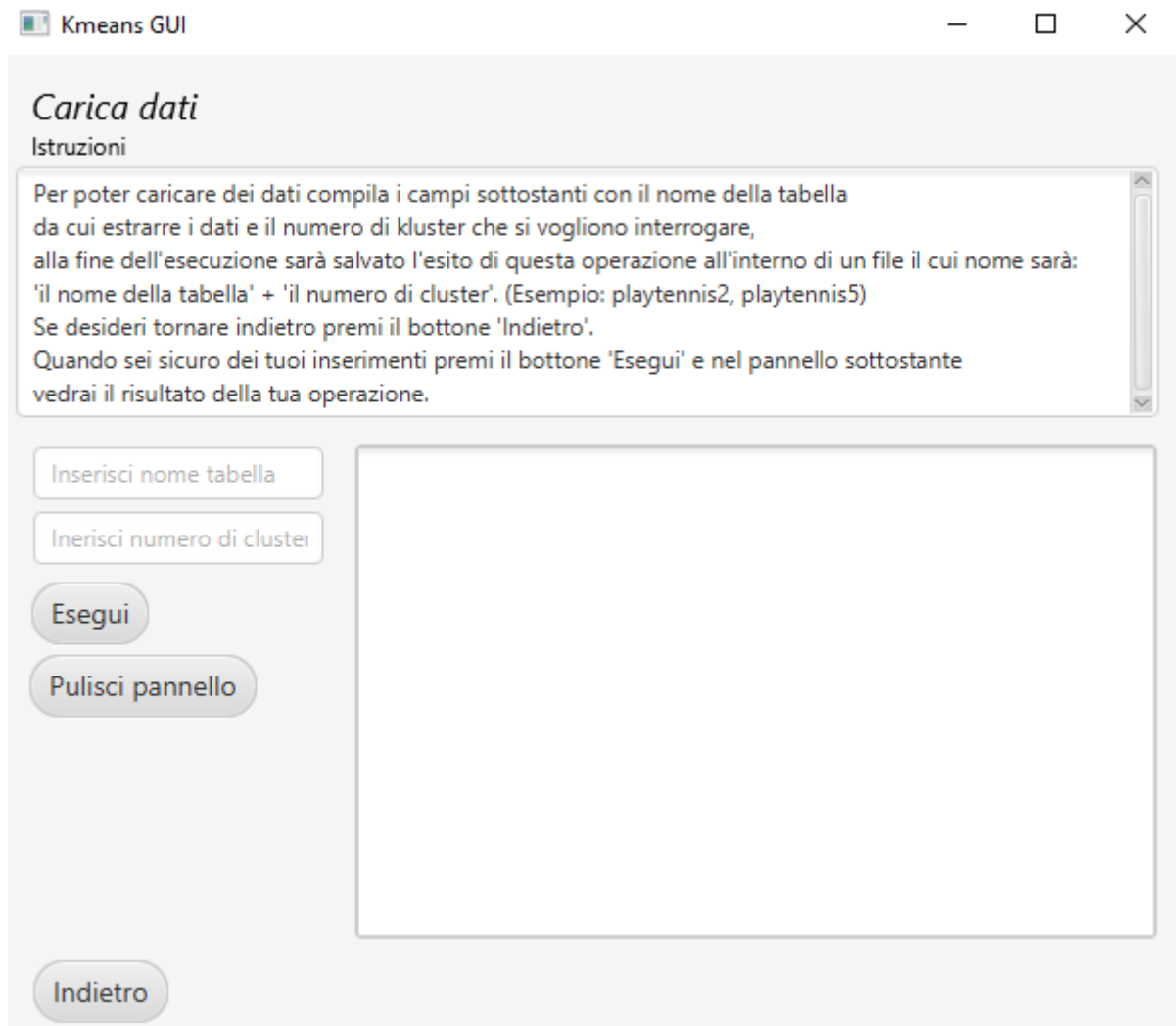


```
C:\Windows\system32\cmd.exe
Started: ServerSocket[addr=0.0.0.0/0.0.0.0,localport=8080]
```

2) Avvio Client:



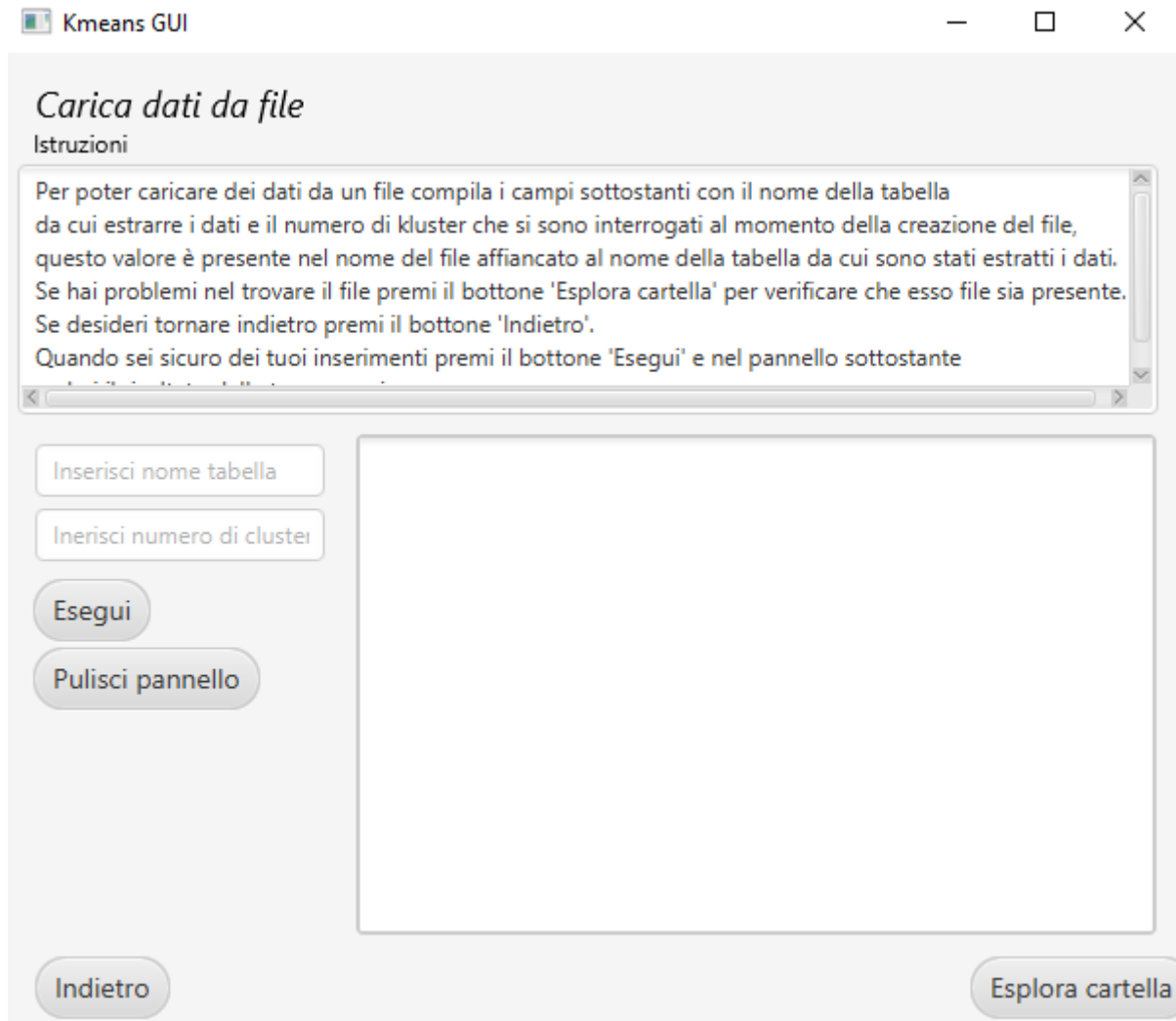
3) Carica dati:



The screenshot shows a window titled "Kmeans GUI" with standard window controls (minimize, maximize, close). The main content area is titled "Carica dati" (Load data) and contains a section labeled "Istruzioni" (Instructions). The instructions text reads: "Per poter caricare dei dati compila i campi sottostanti con il nome della tabella da cui estrarre i dati e il numero di kluster che si vogliono interrogare, alla fine dell'esecuzione sarà salvato l'esito di questa operazione all'interno di un file il cui nome sarà: 'il nome della tabella' + 'il numero di cluster'. (Esempio: playtennis2, playtennis5) Se desideri tornare indietro premi il bottone 'Indietro'. Quando sei sicuro dei tuoi inserimenti premi il bottone 'Esegui' e nel pannello sottostante vedrai il risultato della tua operazione." Below the instructions, there are four input fields: "Inserisci nome tabella", "Inserisci numero di cluster", and two buttons: "Esegui" and "Pulisci pannello". A large empty rectangular box is positioned to the right of these inputs, intended for displaying the results. At the bottom left, there is an "Indietro" (Back) button.

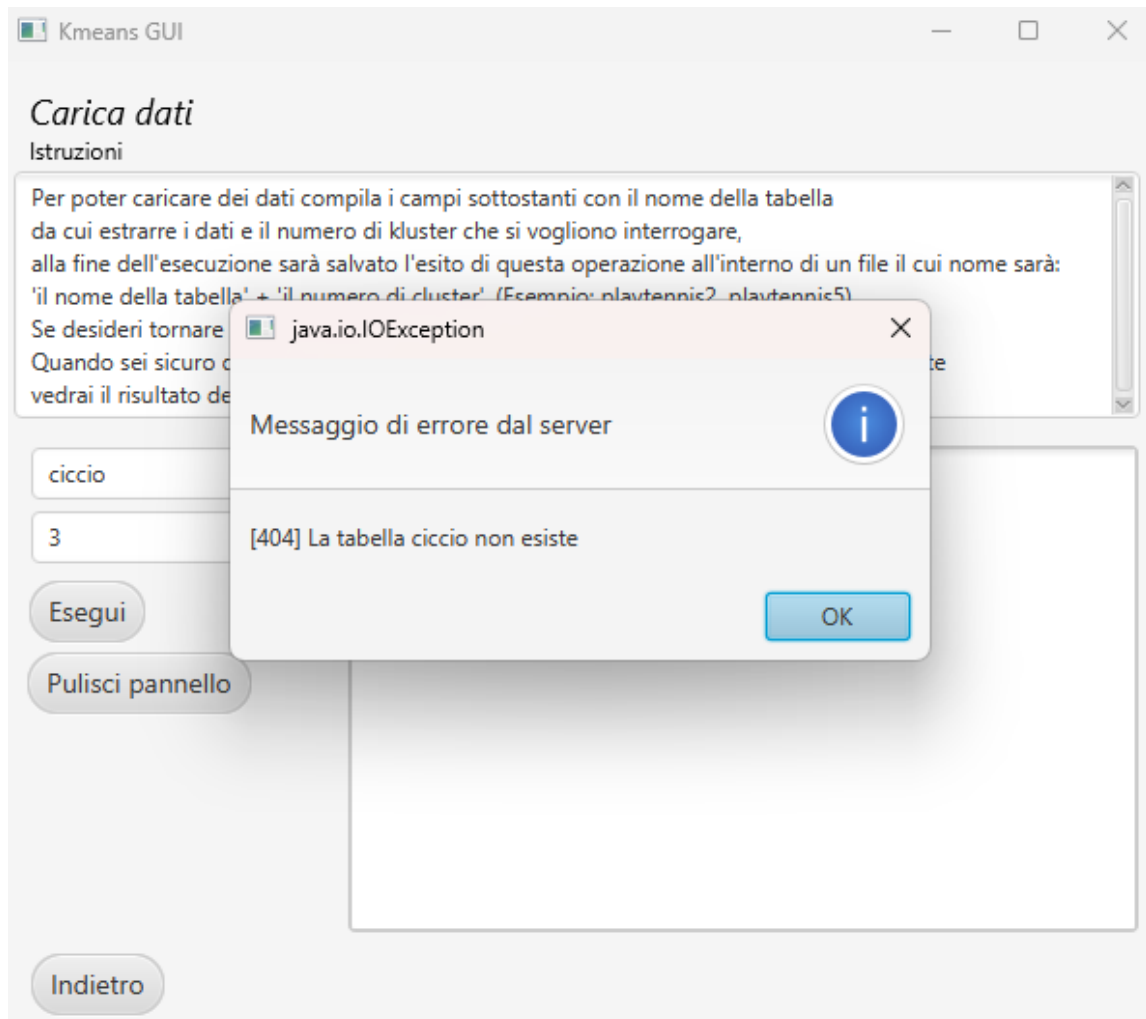
Cliccando su “*Indietro*”, si torna alla schermata iniziale. Cliccando su “*Pulisci pannello*” si svuota il pannello di output dei risultati. Cliccando su “*Esegui*” si inoltra la richiesta al server con gli inserimenti del nome della tabella e il numero di cluster.

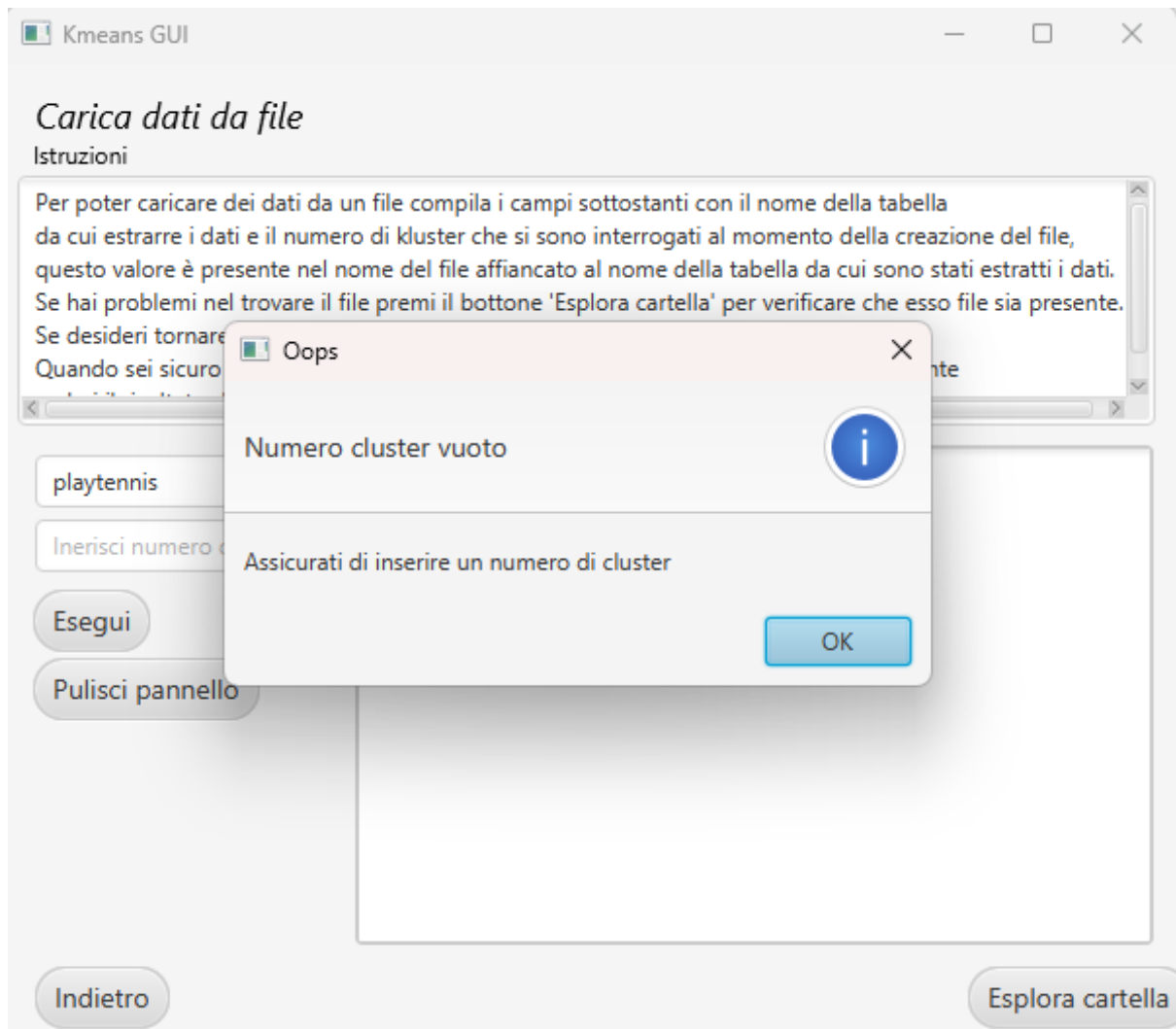
4) Carica dati da file:



Cliccando su “*Indietro*”, si torna alla schermata iniziale. Cliccando su “*Pulisci pannello*” si svuota il pannello di output dei risultati. Cliccando su “*Esplora cartella*” si apre la cartella “File memorizzati”. Cliccando su “*Esegui*” si inoltra la richiesta al server con gli inserimenti del nome della tabella e il numero di cluster.

5) Casi particolari





Per uscire dal programma, premere il tasto chiudi in qualsiasi schermata oppure tornare alla schermata principale e premere il tasto *“Chiudi applicazione”*.

NOTE

ⁱ In alternativa si può aprire il file con un editor di testo e copiare il contenuto nella shell MySQL

ⁱⁱ Nel caso in cui il file jar non funzionasse a causa di librerie non linkate correttamente, è possibile importare queste ultime come librerie esterne al progetto, ed avviare il client direttamente da IDE

ⁱⁱⁱ Per passare dalla versione base a quella estesa o viceversa, assicurarsi di usare la giusta versione del server (\Estesa\). Se necessario, chiudere il server base prima di aprire il server esteso