

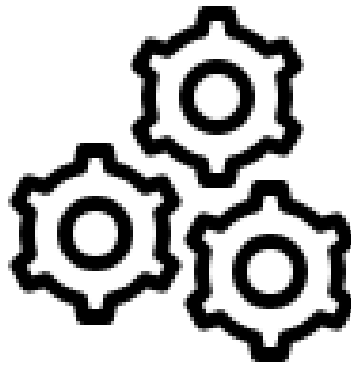
**UNIVERSITÀ DEGLI STUDI DI PADOVA**

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN  
INGEGNERIA INFORMATICA

CORSO DI ELEMENTI DI INGEGNERIA DEL SOFTWARE

## Documento test



**Gruppo di lavoro:**

Giovanni Faedo  
Martino Bissiato  
Marco Facco  
Andrea Maluta

**Anno accademico:**

2023/2024

# Indice

<b>Indice</b>	<b>1</b>
<b>1 Specifiche dei test</b>	<b>2</b>
1.1 Serializzazione e deserializzazione . . . . .	2
1.2 Scaricamento . . . . .	2
1.3 Conteggio dei termini più frequenti con relativo peso . . . . .	3

# Capitolo 1

## Specifiche dei test

Il seguente programma presenta dei test che valutano il corretto comportamento di 3 principali funzionalità:

1. **Serializzazione e deserializzazione** tramite la classe *SerializerDeserializerTest.java*
2. **Scaricamento** tramite la classe *CSVTest.java* e *TheGuardianTest.java*
3. **Conteggio dei termini più frequenti con relativo peso** tramite la classe *WordCounterTest.java*

### 1.1 Serializzazione e deserializzazione

#### SerializerDeserializerTest.java

Questa classe presenta funzioni che testano i processi di serializzazione e deserializzazione. Il metodo *SerializeTest* verifica la corretta creazione di un file JSON. Il test crea un array di 3 articoli e li serializza, creando così un file con estensione json che viene inserito nella cartella “asset”. Il metodo *DeserializerTest* verifica la corretta deserializzazione del file .json precedentemente creato dal metodo *SerializeTest*. Il test verifica la presenza di un file .json nella cartella “asset”, lo deserializza, controlla il risultato della deserializzazione e verifica il contenuto degli articoli. Infine il file .json viene cancellato.

### 1.2 Scaricamento

#### CSVTest.java

La classe presenta 2 funzioni di test: *getArticlesTest()* e *getArticlesWithoutSources()*.

- **getArticlesTest**: verifica che la ricezione degli articoli avvenga correttamente andando a testare il metodo *getArticles()* della classe CSV.java. Il test esegue i seguenti passaggi:

- verifica che chiamando il metodo *getArticles()* non si ottenga un array nullo
- verifica che il titolo del primo articolo di un articolo casuale non sia nullo
- verifica che il body del primo articolo di un articolo casuale non sia nullo
- verifica che la sorgente sia CSV

**getArticlesWithoutSources**: verifica che non possano venire introdotti articoli di cui non si conosce la sorgente (?)

### TheGuardianTest.java

La classe presenta la funzione *getArticlesTest()*, che verifica che avvenga il corretto scaricamento degli articoli dalla fonte “The Guardian”. Il test esegue i seguenti passaggi:

- controlla che *getArticles()* non restituisca il valore null
- controlla che l’array di articoli non sia vuoto dopo avere chiamato il metodo *getArticles()*
- controlla sorgente, body e titolo

## 1.3 Conteggio dei termini più frequenti con relativo peso

### WordCounterTest.java

La classe presenta 4 funzioni: *countTest*; *stopListTest*; *orderTest* e *counterFileToMap*.

- **countTest**: controlla se le parole e la frequenza di esse siano corrette
- **stopListTest**: controlla che le parole da escludere siano effettivamente state immesse alla lista *stopList* e che venga escluso il conteggio di esse.
- **orderTest**: verifica che l’ordine dei termini classificati in base al loro peso sia corretto. Se 2 termini hanno lo stesso peso vengono classificati in ordine alfabetico.
- **counterFileToMap**: è un metodo ausiliario utilizzato per salvare le parole e le frequenze ottenute tramite *WordCounter.getFrequency()*.