



**Istituto Tecnico Tecnologico Statale "Alessandro Volta"**

**Scuol@2.0**

Chimica, Materiali e Biotecnologie  
Grafica e Comunicazione  
Trasporti e Logistica

Via Assisana, 40/E - loc. Piscille - 06135 Perugia  
Centralino 075.31045 fax 075.31046 C.F. 80005450541  
www.avolta.pg.it  
voltauffici@tin.it dirigente@avolta.pg.it

Meccanica, Meccatronica ed Energia  
Elettronica ed Elettrotecnica  
Informatica e Telecomunicazioni

**Corso di**  
**Tecnologie e progettazione di sistemi informatici e di**  
**telecomunicazioni**

**Classi 5AInf, 5BInf, 5CInf a.s.**

**2016/2017**

**Relazione progetto**  
**"Deposito dati"**

**Studente: Nicolò Vescera**

**Docente/i: Monica Ciuchetti**

# Documento di specifica dei Requisiti (SRS)

## 1. Introduzione

Questo programma Java permette di gestire una risorsa condivisa, **Deposito**, in maniera concorrente. Il deposito verrà inizializzato da un **Produttore**, che si occuperà di immettere i dati in un array presente in deposito, ed infine un numero variabile di **Lettori**, in base alla dimensione dell'array, andranno a leggere ed analizzare i dati presenti nella risorsa condivisa.

## 2. Descrizione dell'architettura dell'applicazione (componenti hardware e software)

Questo programma è stato scritto e compilato in un ambiente linux ( Ubuntu 16.10 ), con la versione 8 della JDK e la versione 1.8 della JRE rilasciate da oracle.com. La macchina su cui gira questo sistema è dotata di 8GB di RAM e un processore 8core a 4GHz AMD FX.

### 2.1 Attori

In questa applicazione sono presenti 3 attori: **Lettore**, **Produttore** e il **Sistema**.

Il **Produttore** ha il compito di **popolare** l'array presente nella risorsa condivisa.

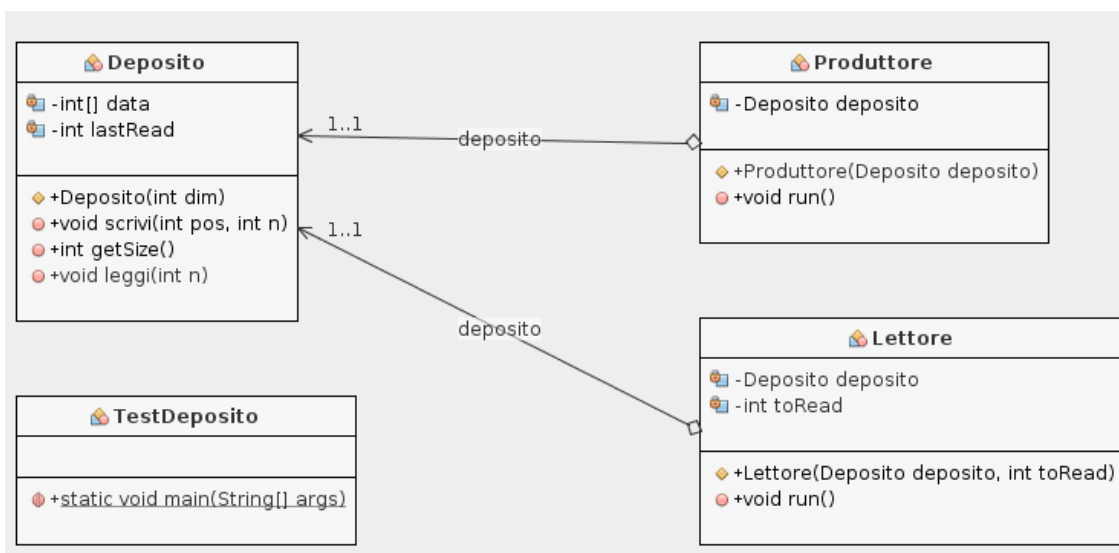
Il **Lettore** deve leggere ed **analizzare** i dati presenti nel deposito. Esisteranno più lettori in base alla quantità di dati da analizzare e ogni lettore leggerà 400 dati alla volta ( un dato è un numero intero ).

Il **Sistema** ha il compito di:

- **Inizializzare** il Deposito con una **dimensione casuale** tra 10000 e 15000,
- **Avviare** il **Produttore**,
- **Istanziare** un numero adeguato di **Lettori**,
- **Avviare** i lettori

# Documento di progettazione ed implementazione dell'applicazione

## 1. Diagramma delle Classi



## **2. Commento al codice**

Il javadoc è nella cartella ./javadoc

## **3. Test dell'applicazione**

Per testare il codice ho utilizzato il seguente metodo: controllare se l'output dell'applicazione era corretto, cioè se tutti i numeri venivano stampati in ordine crescente e non ne veniva omesso nessuno ( es. 10,11,15,16 è un output errato in quanto mancano i numeri dell'11 al 15 ) allora tutto funzionava correttamente. In caso contrario c'era sicuramente un problema nella sincronizzazione dei thread. Il test è stato eseguito più volte per assicurarmi che l'output fosse sempre lo stesso e che l'esecuzione non fosse affidata al caso.

## **Licenza e manuali d'uso**

### **1. Licenza d'uso**

OpenSource

### **2. Manuale d'uso**

Javadoc