

Photogram

# Photo Manager Archiver

Documentazione e manuale

Alessio Colautti & Alex Troilo  
05/12/2025



**Photogram**

# Indice

Caratteristiche e funzionamento (Manuale all'uso) .....	2
Modello ER / Logico (Grafico UML).....	3
Classe PhotoMgrArcv (“main”) .....	4
Classe Owner.....	4
Classe Photo .....	4
Classe Subject.....	4
Classe Condition.....	4
Salvataggio su file.....	5
Conclusioni e sviluppi futuri.....	5
Bibliografia .....	7

# Introduzione

Photo Manager Archiver è un software realizzato in linguaggio Java, con l'ausilio del compilatore Apache NetBeans, ideato per la gestione e l'organizzazione di un archivio di fotografie, con diverse informazioni dettagliate disponibili all'inserimento per ognuna di esse. Successivamente, possono essere visualizzate e analizzate per ogni scopo futuro.

Questo software è stato realizzato da Alessio Colautti e Alex Troilo, due studenti della classe 4B del ramo delle materie informatiche della scuola ITI Galilei Fermi Pacassi di Gorizia.

## Caratteristiche e funzionamento (Manuale all'uso)

All'interno di Photo Manager Archiver, l'utente ha la possibilità, come precedentemente citato, di inserire le informazioni descrittive di un archivio di fotografie e, successivamente, di un numero indefinito di foto in modo da poterne tenere traccia e poterne visualizzare le caratteristiche principali in modo semplice ed intuitivo.

L'utente, all'avvio del programma, può scegliere se interagire dal punto di vista comune, o amministrativo, rendendo così possibile l'utilizzo del suddetto da parte sia dei clienti stessi, sia del possessore dell'archivio. Nel primo caso, l'utente comune è in grado semplicemente di visualizzare le informazioni e le caratteristiche di ogni fotografia, mentre nel secondo caso, all'amministratore è richiesta l'autenticazione con la password (admin) che potrà essere cambiata in futuro dal medesimo nel codice del programma stesso modificando il valore dell'attributo [password] all'inizio del codice. Dopo essersi autenticato, l'amministratore dell'archivio potrà decidere se visualizzare semplicemente le informazioni delle fotografie, aggiungere nuove informazioni di nuove fotografie, o cancellare il file di salvataggio e ricominciare da capo l'archivio.

NB: Una volta inserite le informazioni generali sull'archivio, non potranno essere modificate fino a quando non verrà cancellato il file di salvataggio e verrà ricominciato l'archivio.

Alla richiesta di inserimento delle informazioni, l'utente può decidere:

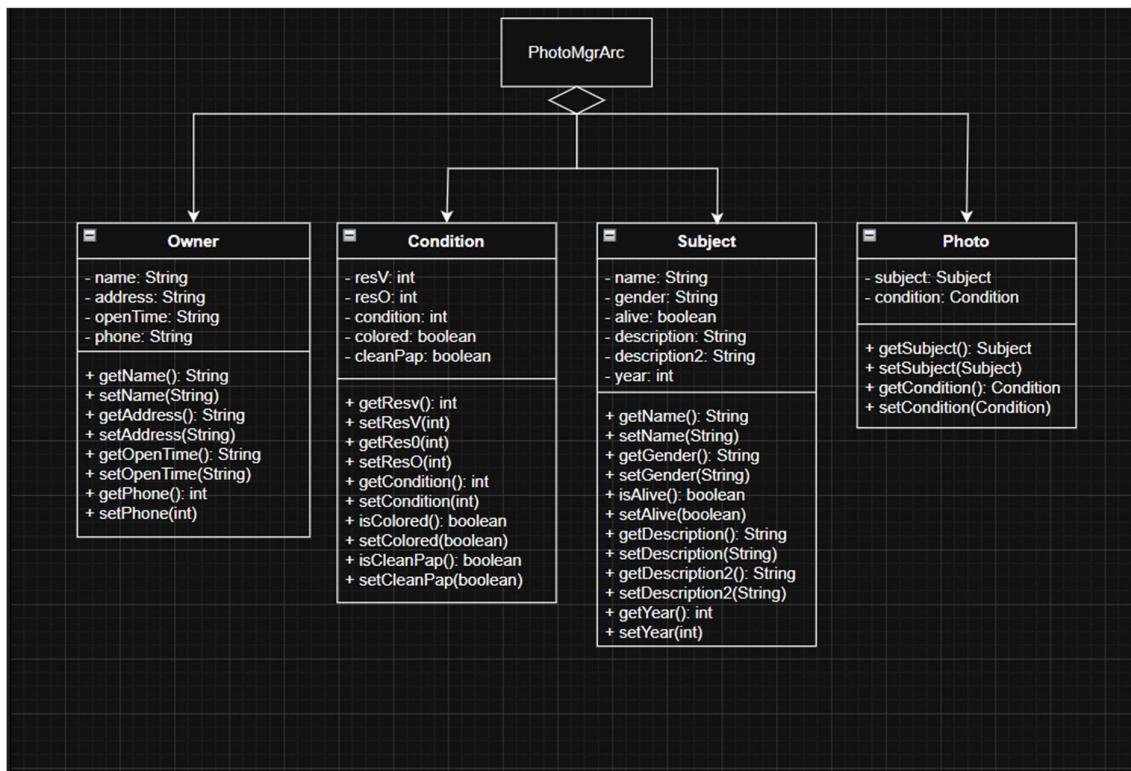
- Il tipo di soggetto presente nella foto
- Le informazioni riguardanti tale soggetto
- Nel caso il soggetto sia una persona, il ruolo della suddetta persona tra quelli proposti
- La descrizione del soggetto nel caso sia un paesaggio, un oggetto o un quadro

- Le informazioni dell'artista, del luogo e dell'anno di realizzazione del soggetto nel caso esso sia un quadro
- Le condizioni della foto fisiche e le informazioni tecniche

Infine, verrà richiesto all'utente se esso vorrà inserire un'altra fotografia, ed in caso di risposta negativa, verranno mostrate le informazioni inserite.

di ricostruire una fotografia: il metodo crea un nuovo Photo, inizializza Subject e Condition, e assegna a ognuno i valori letti dal file, ripristinando così tutte le informazioni tecniche e descrittive.

## Modello ER / Logico (Grafico UML)



Il programma si basa su più classi che lavorano insieme. Ogni classe ha un compito preciso: archivio dei dati, controllo del funzionamento del programma o gestione dell'interazione con l'utente. L'obiettivo è permettere agli utenti normali di consultare le foto e agli amministratori di aggiungere nuove foto, inserire altre informazioni o salvare tutto su file. In questo modo ogni fotografia ha i suoi dati sul soggetto, sullo stato tecnico e sul proprietario dell'archivio, rendendo l'archivio completo e organizzato.

## Classe PhotoMgrArcv (“main”)

PhotoMgrArcv è la classe principale che coordina tutto il programma. Gestisce l’interazione con l’utente, controlla la validità degli input e si occupa di caricare e salvare l’archivio su file. Decide se l’utente è normale o amministratore e, in quest’ultimo caso, permette di aggiungere, visualizzare o cancellare foto. Inoltre crea gli oggetti Photo e ne popola i contenuti con Subject e Condition.

## Classe Owner

La classe Owner rappresenta il proprietario dell’archivio. Memorizza informazioni come nome, indirizzo, orario di apertura e numero di telefono. Serve a dare identità all’archivio e viene utilizzato sia per visualizzare le informazioni sia per salvarle su file.

## Classe Photo

La classe Photo rappresenta una singola fotografia e funge da contenitore di tutte le informazioni relative a quella immagine. Contiene un oggetto Subject, che descrive il soggetto della foto, e un oggetto Condition, che ne descrive lo stato tecnico e fisico. Ogni foto viene aggiunta all’archivio gestito da PhotoMgrArcv.

## Classe Subject

La classe Subject rappresenta ciò che è raffigurato nella fotografia: una persona, un paesaggio, un oggetto o un’opera d’arte. Contiene informazioni come nome, genere, stato di vita, descrizione e anno di creazione, a seconda del tipo di soggetto. Ogni Photo ha il suo Subject specifico per catalogare accuratamente il contenuto.

## Classe Condition

La classe Condition descrive lo stato tecnico e fisico della foto. Include risoluzione, qualità, colore o bianco e nero e tipo di carta. Viene creato per ogni foto e associato all’oggetto Photo, consentendo di avere informazioni complete sullo stato e le caratteristiche della fotografia.

## Salvataggio su file

Nel codice del programma, l'implementazione della scrittura su file dei dati dell'archivio fotografico è stata realizzata attraverso un metodo dedicato, chiamato saveArchiveToFile. Questo metodo ha il compito di esportare su un file esterno tutte le informazioni che l'utente ha inserito: i dati del proprietario dell'archivio e quelli di ogni fotografia registrata. L'obiettivo è permettere al programma di conservare queste informazioni anche dopo la chiusura, così da poterle ricaricare automaticamente al successivo avvio.

Il formato scelto per memorizzare i dati prevede che ogni informazione venga scritta come una riga di testo, con i vari campi separati dal carattere "|". Questo approccio permette sia di leggere il file con facilità sia di ricaricarli successivamente senza errori. La prima riga del file contiene sempre i dati del proprietario dell'archivio, identificati dalla parola chiave OWNER, le righe successive, invece, rappresentano ciascuna una fotografia, e iniziano con la parola PHOTO. In questo modo ogni linea del file mantiene una struttura riconoscibile e facilmente interpretabile dal programma e dall'utente.

Per compilare queste righe, il metodo estrae dalle varie classi (Owner, Subject, Condition) tutti gli attributi necessari e li compila in una stringa unica, che viene poi scritta nel file. Questo permette di salvare tutte le informazioni rilevanti — dalla descrizione del soggetto alle caratteristiche tecniche della foto — in modo compatto e completo.

In continuità con la fase di salvataggio, il caricamento dell'archivio utilizza la stessa struttura del file per ricostruire in memoria tutti i dati salvati in precedenza. Il metodo dedicato apre il file tramite uno Scanner, lo legge riga per riga ed interpreta ciascuna linea grazie alla parola chiave iniziale (OWNER o PHOTO).

Quando viene letta una riga OWNER, i vari campi separati dal carattere "|" vengono estratti e usati per ricreare l'oggetto Owner. Le righe che iniziano con PHOTO, invece, permettono di ricostruire una fotografia: il metodo crea un nuovo Photo, inizializza Subject e Condition, e assegna a ognuno i valori letti dal file, ripristinando così tutte le informazioni tecniche e descrittive.

## Conclusioni e sviluppi futuri

Il programma Photo Manager Archive offre un sistema completo per la gestione di un archivio fotografico, combinando semplicità d'uso e struttura dei dati. Grazie alle classi progettate, ogni fotografia è memorizzata con informazioni sul soggetto, sulle caratteristiche tecniche e sul proprietario dell'archivio, garantendo un'organizzazione chiara e facilmente consultabile. L'interazione con l'utente è guidata e sicura, con controlli sugli input e possibilità di distinguere tra

utenti normali e amministratori. La gestione della persistenza dei dati tramite file permette di conservare tutte le informazioni tra le diverse esecuzioni del programma, rendendolo affidabile e pratico per archivi di piccole e medie dimensioni.

Per quanto riguarda gli sviluppi futuri, il programma presenta numerose possibilità di ampliamento. Una delle principali migliorie al programma potrebbe essere l'introduzione di un'interfaccia grafica così da rendere la comunicazione tra utente e programma più semplice e intuitiva. Inoltre, si potrebbe sostituire il sistema attuale di salvataggio dei dati su file con un metodo più efficace come ad esempio l'utilizzo di un database.

Un'altra possibile evoluzione potrebbe essere la possibilità di ordinare o filtrare le foto in base al tipo di soggetto, tipo di materiale, qualità e altri parametri.

Grazie per la lettura.

***Photogram™***

# Bibliografia

[Creazione file](#)

[Scrittura su file](#)

[Java files](#)

[Try and Catch](#)