



**Progetto di Basi di Dati 2022/2023**  
**Sistema di gestione di una Galleria Fotografica**  
**Geolocalizzata**

**DICEMBRE 2023**

Rippa Andrea	Mirra Gaspare	Liberti Paolo
N86004009	N86004219	N86004255

# Indice

<b>1</b>	<b>Requisiti identificati</b>	<b>3</b>
1.1	Richiesta del lavoro assegnato . . . . .	3
1.2	Funzionalità principali . . . . .	3
1.3	Funzionalità aggiuntive . . . . .	4
<b>2</b>	<b>Progettazione concettuale</b>	<b>5</b>
2.1	Introduzione alla Progettazione . . . . .	5
2.2	Class Diagram e Ristrutturazione . . . . .	5
2.2.1	Analisi delle ridondanze . . . . .	6
2.2.2	Rimozione Generalizzazioni . . . . .	6
2.2.3	Rimozione Attributi Multivalore . . . . .	6
2.2.4	Rimozione Attributi Strutturati . . . . .	6
2.2.5	Accorpamento/Partizionamento di Entità e Associazioni .	6
2.2.6	Identificatore Chiavi Primarie . . . . .	7
2.3	Class Diagram Ristrutturato . . . . .	7
2.3.1	Diagramma Ristrutturato . . . . .	7
<b>3</b>	<b>Dizionari</b>	<b>8</b>
3.1	Dizionario delle Classi . . . . .	8
3.2	Dizionario delle Associazioni . . . . .	10
3.3	Dizionario dei Vincoli . . . . .	13
<b>4</b>	<b>Modello Logico</b>	<b>15</b>
4.1	Schema Logico . . . . .	15
4.2	Funzioni, Procedure e Trigger Functions . . . . .	17
4.2.1	Funzioni e Procedure . . . . .	17
4.2.2	Trigger Functions . . . . .	18

# Capitolo 1

## Requisiti identificati

### 1.1 Richiesta del lavoro assegnato

Il progetto ha come obiettivo la realizzazione di una base di dati in grado di gestire un sistema basato sul raccoglimento di foto geolocalizzate inserite dagli utenti, i quali hanno anche l'opzione di condividerle.

### 1.2 Funzionalità principali

Il sistema è in grado di tracciare ogni foto secondo specifiche caratteristiche tra cui: utente, codice identificativo, dispositivo con cui è stata scattata, luogo, data della fotografia e (se specificato) soggetto. Una foto può contenere più soggetti, che però sono identificati in modo unico nel sistema. L'utente ha la possibilità di osservare la sua galleria personale, che conterrà le sue foto personali, ma può anche partecipare a collezioni condivise con altri utenti, che possono contenere le foto dei partecipanti ad esse. Il sistema può anche gestire la visibilità di ogni foto, andando a specificare se una foto deve essere privata (rendendola invisibile ad utenti esterni) o se deve essere pubblica (rendendola visibile all'interno delle collezioni).

In aggiunta:

- E' possibile eliminare una foto, rendendola indisponibile all'utente originale, ma ancora accessibile se era già presente in una collezione pubblica al momento della cancellazione.
- E' possibile eliminare un utente dal sistema, rimuovendo così tutte le sue foto, fatta eccezione per quelle che contengono come tag altri utenti.
- E' possibile creare un video con le foto nel sistema, dotato di codice unico e caratteristiche come durata e descrizione.

Il sistema, inoltre, può svolgere alcune operazioni di recupero informazioni, come il recupero delle foto per soggetto, per luogo e la top 3 dei luoghi più fotografati.

### **1.3 Funzionalità aggiuntive**

- E' possibile anche accedere ad un'apposita galleria personale dei video.
- La funzionalità di rimozione utente andrà anche a rimuovere i video dal sistema.
- Ogni Collezione Condivisa sarà caratterizzata da un proprio nome, e conterrà anche la data in cui un utente ne è entrato a far parte.

## Capitolo 2

# Progettazione concettuale

### 2.1 Introduzione alla Progettazione

In questa sezione verrà illustrata la progettazione concettuale del lavoro svolto in UML. Verrà poi svolta e illustrata una ristrutturazione che verrà analizzata nella sezione "Ristrutturazione", andando a descrivere le giustificazioni dietro le scelte applicate, che fungeranno da fondamenta per il Class Diagram Ristrutturato.

### 2.2 Class Diagram e Ristrutturazione

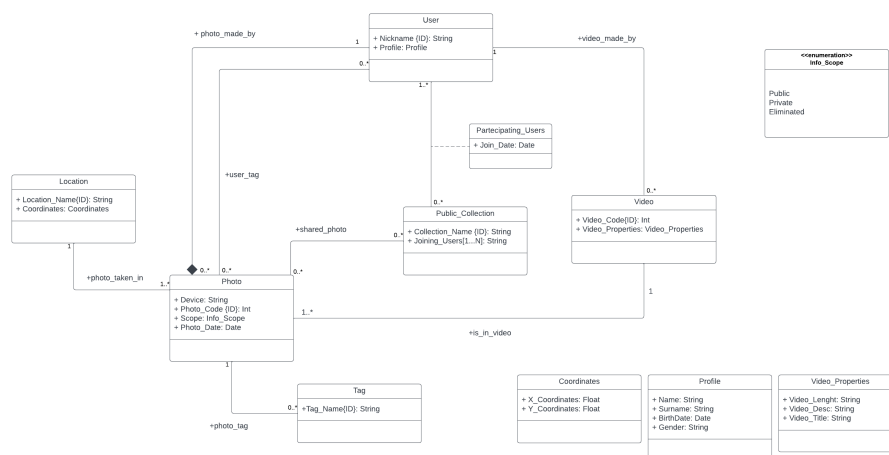


Figura 2.1: Class Diagram in UML non ancora ristrutturato.

Dopo aver analizzato lo schema, possiamo quindi alla ristrutturazione del Class Diagram sopra illustrato, andando a definire nel dettaglio le motivazioni e le caratteristiche per ogni passaggio svolto.

### 2.2.1 Analisi delle ridondanze

Durante l'analisi delle ridondanze, è stato visto come, l'aggiunta di un attributo per il conteggio delle foto per luogo renda più efficiente le operazioni di recupero dei luoghi. Si è quindi presa la decisione di aggiungere l'attributo **"Photo Count"** in **"Location"**

### 2.2.2 Rimozione Generalizzazioni

Vista l'assenza di entità padri e figlie, non è stato necessario eliminare generalizzazioni.

### 2.2.3 Rimozione Attributi Multivalore

Nel modello non ristrutturato è presente un attributo multivalore, cioè **"Joining Users"**.

Nella ristrutturazione, è stato deciso di rimuovere completamente tale attributo, sfruttando invece l'associazione tra **"User"** e **"Public Collection"** per andare a tenere traccia degli utenti che partecipano a tale collezione.

### 2.2.4 Rimozione Attributi Strutturati

Nel modello non ristrutturato sono presenti più attributi strutturati, ovvero:

- **"Coordinates"**
- **"Profile"**
- **"Video Properties"**

Per ognuno di questi è stato deciso di estrarre tali attributi nelle rispettive entità.

### 2.2.5 Accorpamento/Partizionamento di Entità e Associazioni

Non è stato necessario svolgere operazioni di accorpamento o partizionamento.

## 2.2.6 Identificatore Chiavi Primarie

Sono stati scelti i seguenti attributi per assegnare le rispettive chiavi primarie:

Entità	Chiave Primaria
User	Nickname
Photo	Photo Code
Location	Location Name
Video	Video Code
Public Collection	Collection Name
Tag	Tag Name

## 2.3 Class Diagram Ristrutturato

### 2.3.1 Diagramma Ristrutturato

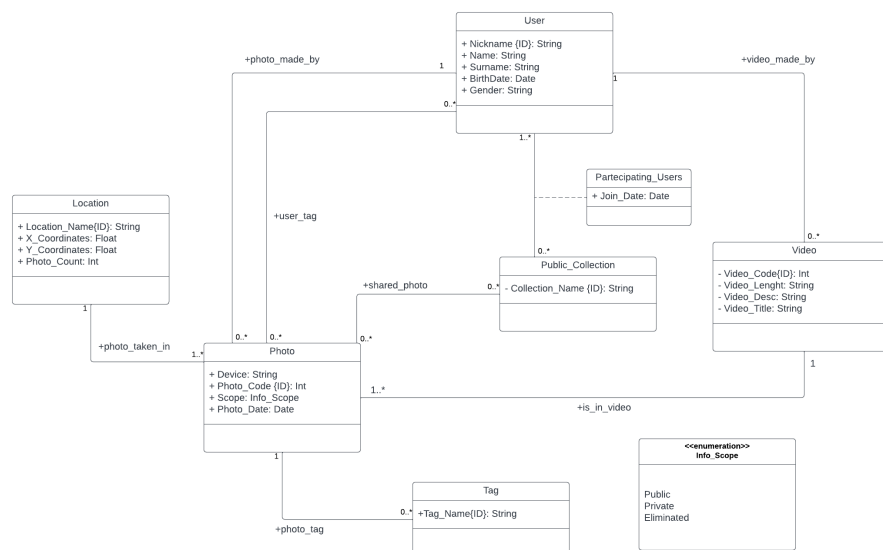


Figura 2.2: Class Diagram in UML Ristrutturato.

## Capitolo 3

# Dizionari

### 3.1 Dizionario delle Classi

PUBLIC COLLECTION	
<b>DESCRIZIONE:</b> Collezione di foto condivise dagli utenti.	<b>ATTRIBUTI:</b> <b>Collection_Name(String):</b> Nome identificativo univoco della galleria pubblica.

VIDEO	
<b>DESCRIZIONE:</b> Sequenza di foto scelte dall'utente che vanno a formare un video.	<b>ATTRIBUTI:</b> <b>Video_Code(Int):</b> Codice identificativo del video. <b>Video_Lenght(String):</b> Lunghezza del video (Formato [hh]-[mm]-[ss]). <b>Video_Desc(String):</b> Breve descrizione del video. <b>Video_Title(String):</b> Titolo dato dall'utente al video.



PHOTO	
<b>DESCRIZIONE:</b> Fotografia scattata e caricata dall'utente.	<b>ATTRIBUTI:</b> <b>Device(String):</b> Dispositivo utilizzato per lo scatto della fotografia.  <b>Photo.Code(Int):</b> Codice Identificativo dello scatto.  <b>Scope(Info.Scope):</b> Stato della fotografia (public,private o eliminated)  <b>Photo.Date(Data):</b> Data dello scatto della fotografia (Formato DD/MM/YYYY).

LOCATION	
<b>DESCRIZIONE:</b> Luogo in cui è stata scattata la fotografia	<b>ATTRIBUTI:</b> <b>Location_Name(String):</b> Nome Identificativo della località di scatto della fotografia.  <b>X_Coordinates(Float):</b> Coordinate di Latitudine geografica del luogo dello scatto.  <b>Y_Coordinates(Float):</b> Coordinate di Longitudine geografica del luogo dello scatto.  <b>Photo.Count(Int):</b> Counter del luogo che viene utilizzato per la classifica dei luoghi più immortalati.

TAG	
<b>DESCRIZIONE:</b> Tipo di soggetto che può avere una fotografia.	<b>ATTRIBUTI:</b> <b>Tag_Name(String):</b> Nome identificativo del soggetto della fotografia.

UTENTE	
<b>DESCRIZIONE:</b> Raccolta delle informazioni dell'utente	<b>ATTRIBUTI:</b> <b>Nickname(String):</b> Nome identificativo scelto dell'utente.  <b>Name(String):</b> Nome dell'utente.  <b>Surname(String):</b> Cognome dell'utente.  <b>BirthDate(Date):</b> Data di nascita dell'utente(Formato DD/MM/YYYY).  <b>Gender(String):</b> Sesso dell'utente.

### 3.2 Dizionario delle Associazioni

user_tag	
<b>DESCRIZIONE:</b> Relazione tra le fotografie e gli utenti presenti nelle stesse.	<b>CLASSI:</b> <b>User[0..*]:</b> Uno o più utenti presenti nella fotografia.  <b>Photo[0..*]:</b> Una o più fotografie nelle quali sono presenti altri utenti.

video_made_by	
<b>DESCRIZIONE:</b> Relazione che definisce l'appartenza di un video ad un utente.	<b>CLASSI:</b> <b>USER[1]:</b> Utente creatore del video.  <b>Video[0..*]:</b> Video creati dall'utente.

is_in_video	
<b>DESCRIZIONE:</b> Relazione che definisce la raccolta di foto che vanno a comporre il video.	<b>CLASSI:</b> <b>Photo[1..*]:</b> Raccolta di fotografie che vanno a comporre il video.  <b>Video[1]:</b> Sequenza di fotografie scelte dall'utente.

photo_tag	
<b>DESCRIZIONE:</b> Relazione che definisce i soggetti presenti nella fotografia.	<b>CLASSI:</b> <b>Photo[1]:</b> Fotografia in cui sono presenti i soggetti.  <b>Tag[0..*]:</b> Tag dei soggetti presenti nella fotografia.

shared_photo	
<b>DESCRIZIONE:</b> Relazione che definisce quali fotografie sono presenti nella galleria pubblica.	<b>CLASSI:</b> <b>Photo[0..*]:</b> Fotografie inserite nella galleria pubblica.  <b>Public_Collection[0..*]:</b> Galleria pubblica in cui sono raccolte le fotografie condivise dagli utenti.

photo_made_by	
<b>DESCRIZIONE:</b> Relazione che definisce l'appartenenza di una o più fotografie ad un utente.	<b>CLASSI:</b> <b>User[1]:</b> Utente autore della fotografia.  <b>Photo[0..*]:</b> Una o più fotografie scattate dall'utente.

photo_taken_in	
<b>DESCRIZIONE:</b> Relazione che definisce la località in cui sono state scattate una o più fotografie.	<b>CLASSI:</b> <b>Photo[1..*]:</b> Una o più foto scattate in una determinata località.  <b>Location[1]:</b> Località in cui sono state scattate una o più fotografie.

participating_users	
<b>DESCRIZIONE:</b> Relazione che definisce gli utenti che partecipano ad una galleria pubblica	<b>ATTRIBUTI:</b> <b>Join_Date(Date):</b> Data in cui l'utente è entrato a far parte della galleria pubblica  <b>CLASSI:</b> <b>User[1..*]:</b> Utenti che partecipano alla galleria pubblica  <b>Public_Collection[0..*]:</b> Galleria pubblica a cui gli utenti partecipano

### 3.3 Dizionario dei Vincoli

Fullname
<b>DESCRIZIONE:</b> Vincolo di dominio che assicura l'uso di caratteri esclusivamente alfabetici durante l'inserimento di nome e cognome dell'utente.

Check_Gender
<b>DESCRIZIONE:</b> Vincolo di check che assicura l'inserimento di gender 'M' o 'F' (rispettivamente maschio o femmina).

Delete_Photo
<b>DESCRIZIONE:</b> Vincolo che assicura, dopo aver eliminato una foto, che tale rimanga comunque in una collezione pubblica andando a non mostrare l'utente che ha scattato tale foto.

Private_Photo
<b>DESCRIZIONE:</b> Vincolo che assicura, a seguito di un inserimento/aggiornamento di una foto privata, che tale foto non appaia in collezioni pubbliche.

Public_Photo
<b>DESCRIZIONE:</b> Vincolo che, forzando l'inserimento di una foto privata in una collezione pubblica, la fa diventare pubblica.

<b>Add_User_Collection</b>
<b>DESCRIZIONE:</b> Vincolo che aggiunge l'utente alla lista di partecipanti della collezione pubblica nel caso esso non ne faccia già parte.

<b>Delete_User</b>
<b>DESCRIZIONE:</b> Vincolo che gestisce i dati a seguito di una eliminazione di un utente assicurandosi che tutti i suoi contenuti (come foto o video) siano eliminati fatta eccezione per le foto dove è taggato un altro utente.

## Capitolo 4

# Modello Logico

### 4.1 Schema Logico

**PHOTO**(Photo\_Code, Device, Scope, Photo\_Date)

**TAG**(Tag\_Name)

**USER**(Nickname, Name, Surname, BirthDate, Gender)

**LOCATION**(Location\_Name, X\_Coordinates, Y\_Coordinates, Photo\_Count)

**VIDEO**(Video\_Code, Video\_Lenght, Video\_Desc, Video\_Title)

**PUBLIC\_COLLECTION**(Collection\_Name)

**IS\_IN\_VIDEO**(Video\_Code, Photo\_Code)

IS\_IN\_VIDEO.Video\_Code  $\leftarrow$  *VIDEO.Video\_Code*

IS\_IN\_VIDEO.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

**PARTECIPATING\_USERS**(Join\_Data, Collection\_Name, Nickname)

PARTECIPATING\_USERS.Collection\_Name  $\leftarrow$  *PUBLIC\_COLLECTION.Collection\_Name*

PARTECIPATING\_USERS.Nickname  $\leftarrow$  *USER.Nickname*

**VIDEO\_MADE\_BY**(Video\_Code, Nickname)

VIDEO\_MADE\_BY.Video\_Code  $\leftarrow$  *VIDEO.Video\_Code*

VIDEO\_MADE\_BY.Nickname  $\leftarrow$  *USER.Nickname*

**USER\_TAG**(Nickname, Photo\_Code)

USER\_TAG.Nickname  $\leftarrow$  *USER.Nickname*

USER\_TAG.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

**PHOTO\_TAG**(Tag\_Name,Photo\_Code)

PHOTO\_TAG.Tag\_Name  $\leftarrow$  *TAG.Tag\_Name*

PHOTO\_TAG.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

**PHOTO\_TAKEN\_IN**(Location\_Name,Photo\_Code)

PHOTO\_TAKEN\_IN.Location\_Name  $\leftarrow$  *LOCATION.Location\_Name*

PHOTO\_TAKEN\_IN.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

**PHOTO\_MADE\_BY**(Nickname,Photo\_Code)

PHOTO\_MADE\_BY.Nickname  $\leftarrow$  *USER.Nickname*

PHOTO\_MADE\_BY.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

**SHARED\_PHOTO**(Photo\_Code,Collection\_Name)

SHARED\_PHOTO.Photo\_Code  $\leftarrow$  *PHOTO.Photo\_Code*

SHARED\_PHOTO.Collection\_Name  $\leftarrow$  *PUBLIC\_COLLECTION.Collection\_Name*



## 4.2 Funzioni, Procedure e Trigger Functions

### 4.2.1 Funzioni e Procedure

#### **rendi\_foto\_privata**

Procedura che aggiorna la visibilità di una foto a 'Private'.

#### **rendi\_foto\_pubblica**

Procedura che aggiorna la visibilità di una foto a 'Public'.

#### **elimina\_foto**

Procedura che permette l'eliminazione di una foto.

#### **galleriapersonale()**

Funzione che permette all'utente di visualizzare le proprie foto.

#### **galleriapersonalevideo()**

Funzione che permette all'utente di visualizzare i propri video.

#### **foto\_stesso\_soggetto()**

Funzione che permette di visualizzare tutte le foto che hanno in comune lo stesso soggetto.

#### **foto\_stesso\_luogo()**

Funzione che recupera tutte le foto che condividono lo stesso luogo.

#### **top\_3\_luoghi()**

Funzione che permette di visualizzare le 3 località più immortalate.

#### **video\_foto()**

Funzione che permette di visualizzare tutte le singole foto che compongono un determinato video.

### 4.2.2 Trigger Functions

#### **Delete.Photo()**

Trigger function che imposta lo scope a 'Eliminated' delle fotografie che sono rimosse dal database.

#### **Delete.User()**

Trigger function che gestisce i dati di un utente eliminato dal database.

#### **Location.Count()**

Trigger function che viene attivata ad ogni inserimento di una nuova foto incrementando il counter della location associata.

#### **Location.Count.Subtract()**

Trigger function che viene attivata ad ogni rimozione di una foto diminuendo il counter della location associata.

#### **NewUser.Collection()**

Trigger function che viene attivata nel momento in cui viene inserita una foto in una galleria condivisa aggiungendo l'utente tra la lista dei partecipanti se non vi è presente.

#### **Private.Photo()**

Trigger function che viene attivata a seguito di una modifica di una foto rendendo lo scope 'Private' ed eliminando la foto dalle gallerie condivise.

#### **Public.Photo()**

Trigger function che viene attivata a seguito della condivisione di una foto modificando lo scope da 'Private' a 'Public'.