Gallery management system to store, organize and retrieve images

A picture containing graphical user interface

Description automatically generated

**COURSE TITLE**

Database Foundations for Business Analytics

**INSTRUCTOR**

Kannan Srikanth

**Business Requirement:**

We click thousands of pictures every day with friends, family, relatives, during vacations, etc. With lots of pictures in our gallery, it becomes quite difficult to retrieve the images we need. It takes much time to search for that one specific image that we want to look at. This is when the Photo Collage Management System comes into the picture. This tool helps users store, organize, and retrieve images when needed without having to search manually.

Features:

* Pictures are stored in the database based on a few pointers such as location, date, event, importance, and comments.
* Retrieval is done using these pointers, i.e., searching is done when the user inputs any or all the pointers mentioned above.

Existing Challenges:

* Difficult to Retrieve: It becomes difficult to retrieve images when they are unorganized and even more difficult if it is a large gallery.
* Time-Consuming: A typical large gallery includes pictures of many events such as vacations, celebrations, etc. It is time-consuming to search for a specific image of a particular event.

**Assumptions:**

* For this project we have used the data of a person’s gallery.
* Every user has a user id and password which helps in logging in the user.
* There are no missing values in the dataset.
* Many categories are used which helps in easy retrieval like location, event, gallery etc.

**Design:**

There are seven tables that were used to build the photo collage management system. Database tables are as follow:

1. Image: All the details regarding a particular image are stored in this table. Primary key in this table is image id.
2. User: It contains attributes like user id, password, email, contact number etc. The primary key in this able is user id and the foreign key is gallery id which is taken from gallery table. The attributes user id and password are used to login initially.
3. Event: This table contains the event name, event type, and event id. Here, the primary key is event id, and the foreign key is image id which references image table.
4. Location: It contains the geographical information where the picture is taken like location name, city, state, country, and zip code. Image id acts as the foreign key which references the image table and location id is the primary key which helps in easy retrieval.
5. Gallery: Columns like gallery title, description and type are stored in this table. It shows us the information of the gallery and description for that gallery etc. Gallery id is the primary key and event id the foreign key which is taken from event table.
6. Reaction Table: It has the type of reaction given to an image and their description. Reaction id is the primary key and image id is the foreign key.
7. Technical Details: It contains all the information regarding the device that has been used for capturing the images. Details like device type, its megapixels, image resolution, and image size in kilobytes are present in this table. Device id is the primary key and image id is the foreign key which references from image table.

**Entity – Relationship Diagram:**

Graphical user interface, application

Description automatically generated

**Table Names:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Primary key** | **Foreign key** | **No. of rows** |
| User | User ID | Gallery ID | 30 |
| Image | Image ID | - | 30 |
| Event | Event ID | Image ID | 100 |
| Location | Location ID | Image ID | 100 |
| Gallery | Gallery ID | Event ID | 3 |
| Reaction | Reaction ID | Image ID | 100 |
| Technical Details | Device ID | Image ID | 10 |

**Relationships:**

|  |  |  |
| --- | --- | --- |
| **Table 1** | **Table 2** | **Relationship** |
| Gallery | User | 1:n |
| Image | location | 1:n |
| Image | Reaction | 1:n |
| Image | Technical details | 1:n |
| Image | Event | 1:n |
| Event | Gallery | 1:n |

**Database Tables:**

**Graphical user interface, text, application, chat or text message

Description automatically generated Graphical user interface, text, application, chat or text message

Description automatically generated Graphical user interface, text, application, chat or text message

Description automatically generated**

**Graphical user interface, text, application, chat or text message

Description automatically generated Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application, table

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated**

**Data:**

Data for image table:

**Table

Description automatically generated**

Data for location table:

Table

Description automatically generated

Data for event:

**Graphical user interface, text, application

Description automatically generated**

Data for Gallery:

**Graphical user interface, application

Description automatically generated**

Data for reaction:

**Graphical user interface

Description automatically generated with medium confidence**

Data for technical\_details:

Table

Description automatically generated

**Conclusion:**

Due to the unstructured and vast size of the current gallery management system, retrieving photographs becomes difficult. If the gallery has a lot of events, it can take a long time to find an image from one of them.

Pictures are sorted when they are stored with the help of the photo collage management system. As a result, retrieving photos using pointers placed at the start becomes easy. It's no longer difficult to find photographs of specific events, such as 24th birthday parties, because users may search directly using an event or date filter. Thousands of photographs from earlier years or events can be stored and retrieved by large organizations such as educational institutions and event management firms. This is an easy-to-use interface that just requires a few search parameters to find the photographs you need.