

BLOG MANAGEMENT SYSTEM

BY: NIKHIL KUMAR GAUTAM
(U18CO053)

DEFINITION:

This project implements Database Management for a basic Blogging Website.

It has four entities:

1. Blog
2. User
3. Comments
4. Likes

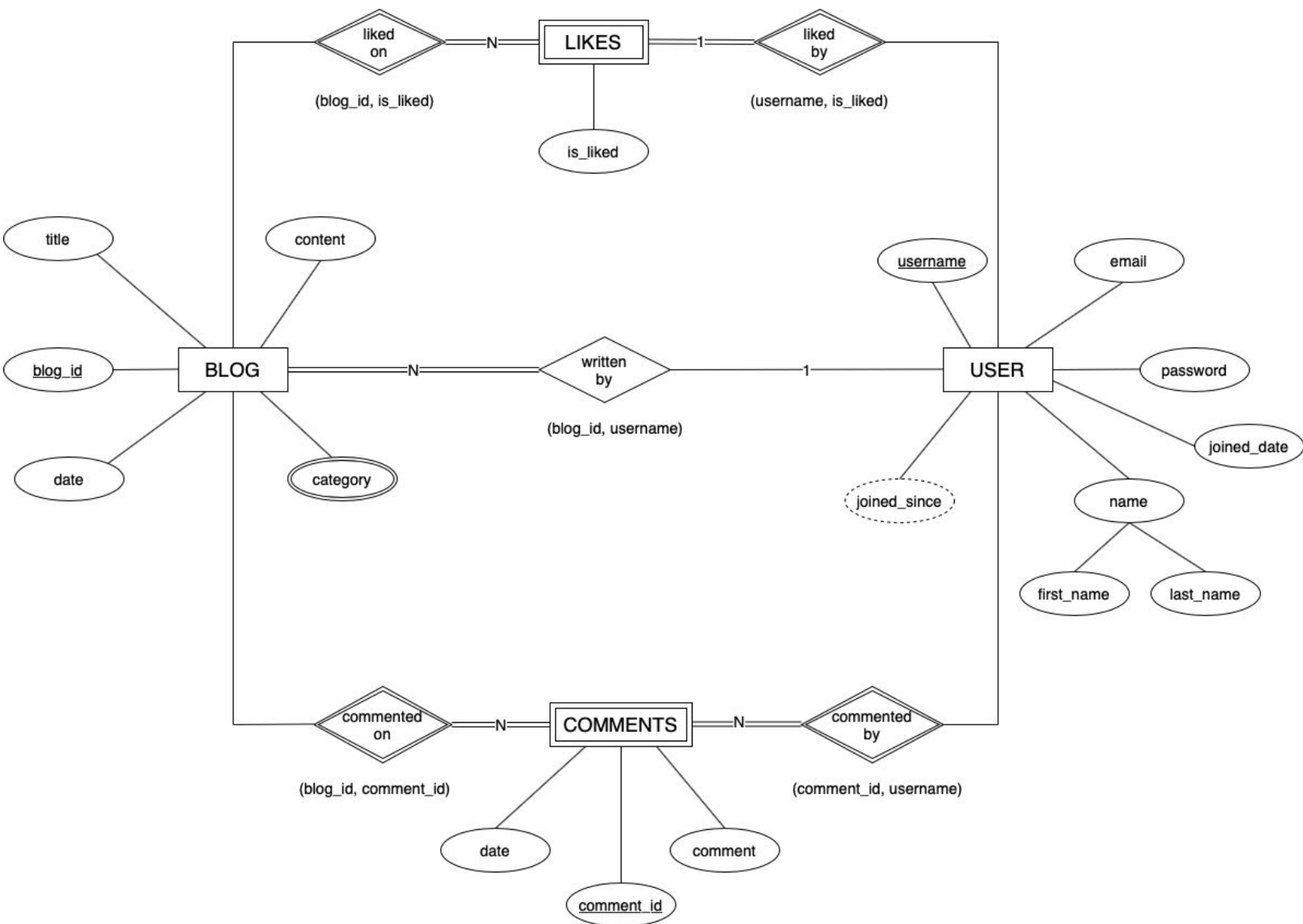
Blog - Blog will be written by a user. A blog will have blog_id (unique id), title, content, published date, category to which it belongs like sports, technology, etc. Blog can be written by all users.

User - A user can write any number of blogs. A user will have a username, email, password, joined_date, name which further divides into first name and last name, joined_since can be derived from joined_date.

Comments - User can comment any number of times on any blog. Comments consist of comment_id, comment data and date.

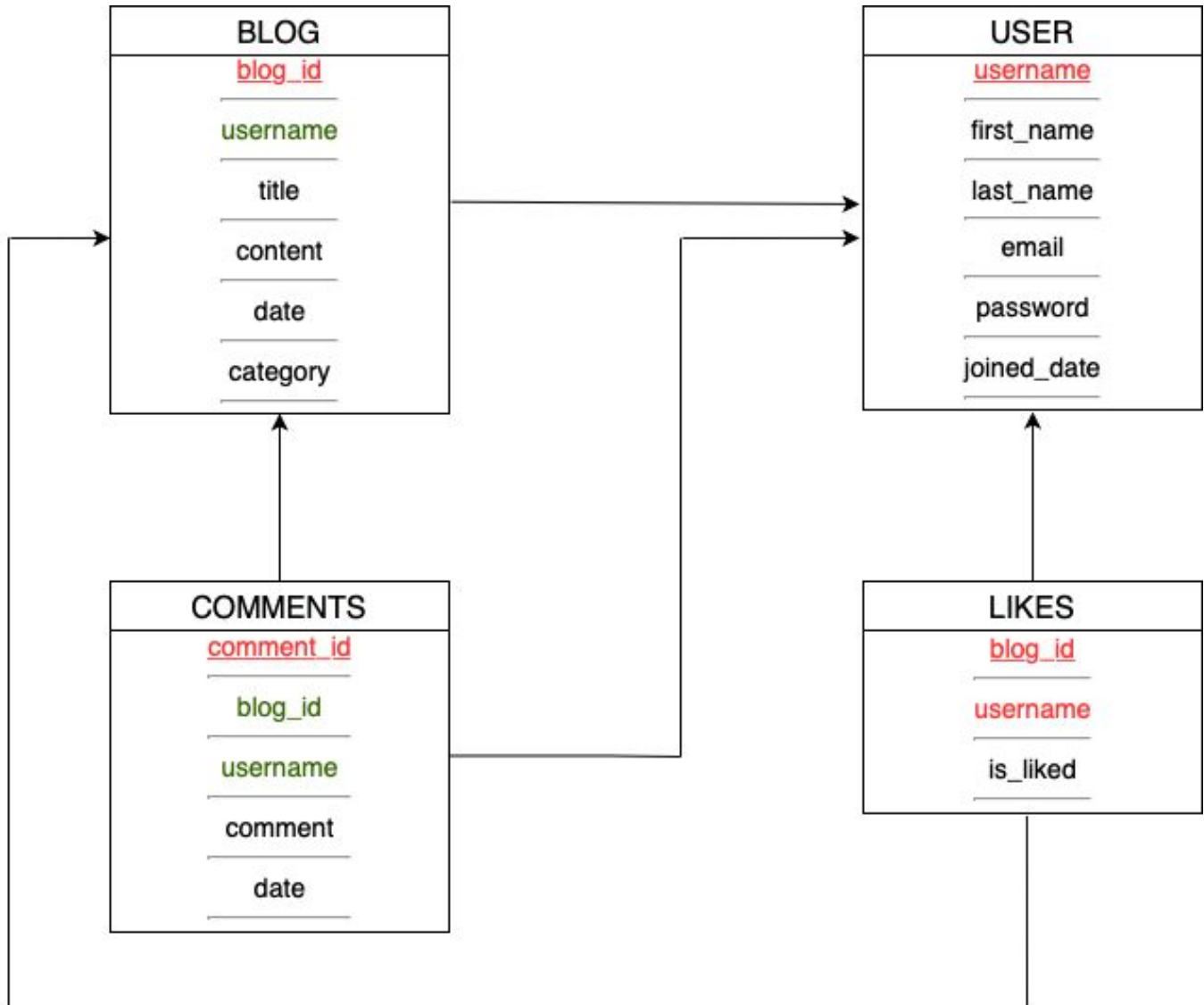
Likes - Users can like any blog but only one like to a particular blog. A blog can be liked by any number of users.

ENTITY-RELATIONSHIP MODEL DIAGRAM



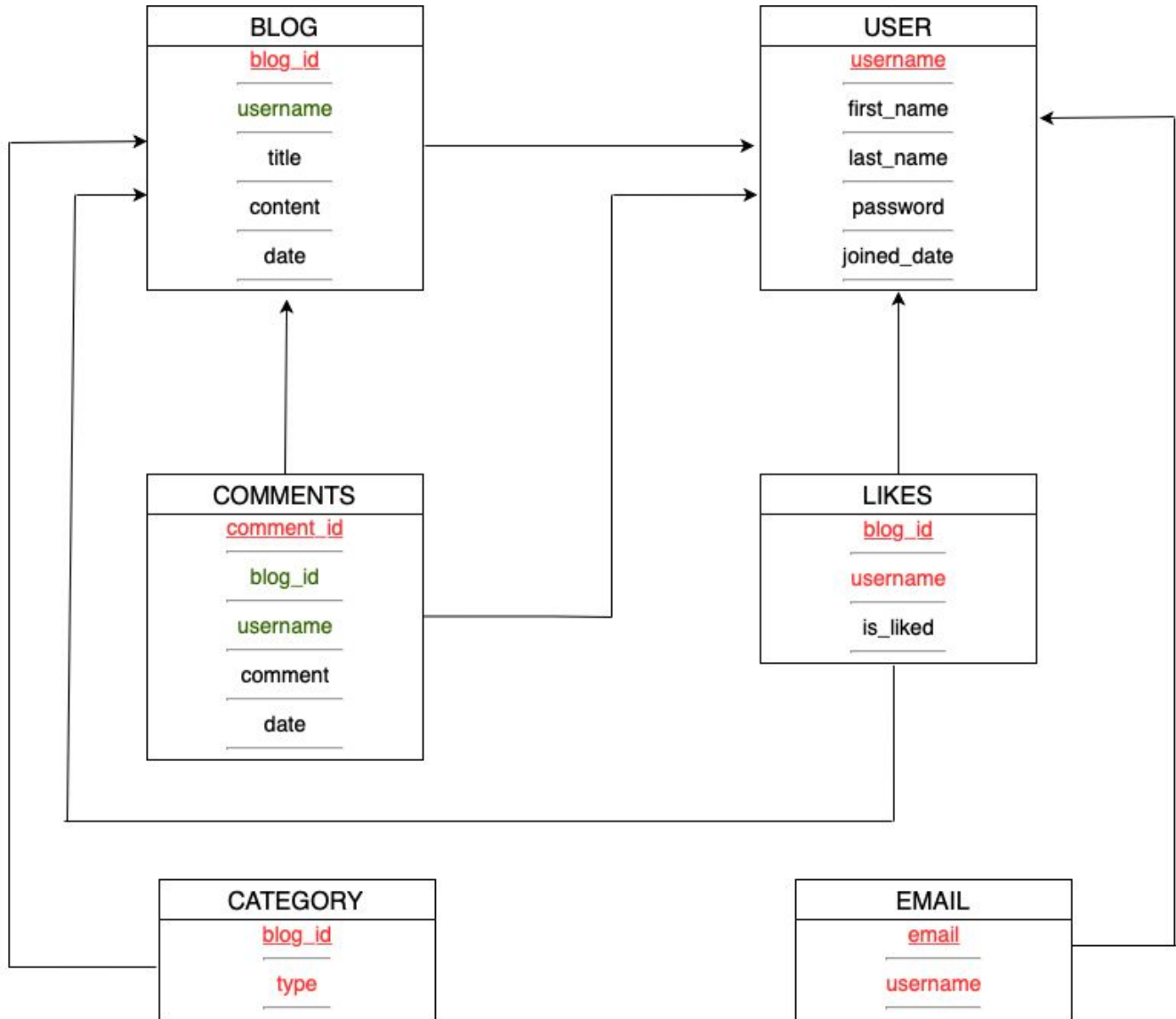
DATABASE SCHEMA

(Without Normalization)



DATABASE SCHEMA

(Normalized)



NORMALIZATION

1NF:

- All attributes are unique.
- There is no multi valued attribute in any of the Database Schema.

Therefore, 1NF is satisfied.

2NF:

- All relations are in 1NF.
- Partial Dependency:
 - The relations 'BLOG', 'USER', 'COMMENTS' do not have a composite key as a primary key.
 - The relations 'CATEGORY' and 'EMAIL' have only prime attributes.
 - In relation 'LIKES', none of the two attributes can identify an attribute.

Hence, there is no Partial Dependency in any of the relations.

Therefore, 2NF is satisfied.

3NF:

- All relations are in 2NF.
- Transitive Dependency:
 - The relations 'CATEGORY', 'EMAIL', 'LIKES' do not have more than one non-prime attribute so they don't have a Transitive Dependency.
 - In relation 'BLOG', a user can post more than one blog. A user can also post two blogs with the same content, same title and on same date. So, none of the attributes can determine other non-prime attributes.

- In relation 'USER', it is possible that two person with same name can register on the same day and they choose the same password.
So, none of the attributes can determine other non-prime attributes.
- In relation 'COMMENTS', it is possible that a user makes same comment multiple times on the same blog, on the same day.
So, none of the attributes can determine other non-prime attributes.

Hence, there is no Transitive Dependency in any of the relations.

Therefore, 3NF is satisfied.

BCNF:

- All relations are in 3NF.
- In BCNF, only those Functional Dependencies are allowed which are of the form Prime Attribute (PA) \rightarrow Non-Prime Attribute (non-PA).
 - In relations 'CATEGORY' and 'EMAIL', non-prime attributes are not present.
So, all FD present will be of the form PA \rightarrow non-PA.
 - In relation 'BLOG', all the non-prime attributes 'username', 'title', 'content', 'date' can be the same for two different 'blog_id'.

So, no non-prime attributes (either collectively or individually) can determine the prime attribute 'blog_id'.

Only FD that exists is:

$\text{blog_id} \rightarrow \text{username, title, content, date}$

- In relation 'USER', all the non-prime attributes 'first_name', 'last_name', 'password', 'joined_date' can be the same for two different 'username'.

So, no non-prime attributes (either collectively or individually) can determine the prime attribute 'username'.

Only FD that exists is:

$\text{username} \rightarrow \text{first_name}, \text{last_name}, \text{password}, \text{joined_date}$

- In relation 'COMMENTS', all the non-prime attributes 'comment', 'date', 'blog_id', 'username' can be the same for two different 'comment_id' as the same person can write the same comment on a blog multiple times.

So, no non-prime attributes (either collectively or individually) can determine the prime attribute 'comment_id'.

Only FD that exists is:

$\text{comment_id} \rightarrow \text{comment}, \text{date}, \text{blog_id}, \text{username}$

- In relation 'LIKES', the non-prime attribute 'is_liked' cannot determine any prime attribute because many blogs can be liked or unliked in the same state of relation.

So, a non-prime attribute cannot determine the prime attribute 'blog_id', 'username'.

Only FD that exists is:

$\text{blog_id}, \text{username} \rightarrow \text{is_liked}$

Hence, all FD are of the type $\text{PA} \rightarrow \text{non-PA}$

Therefore, BCNF is satisfied.

THANK YOU