

**Database Management Systems
(DBMS)**

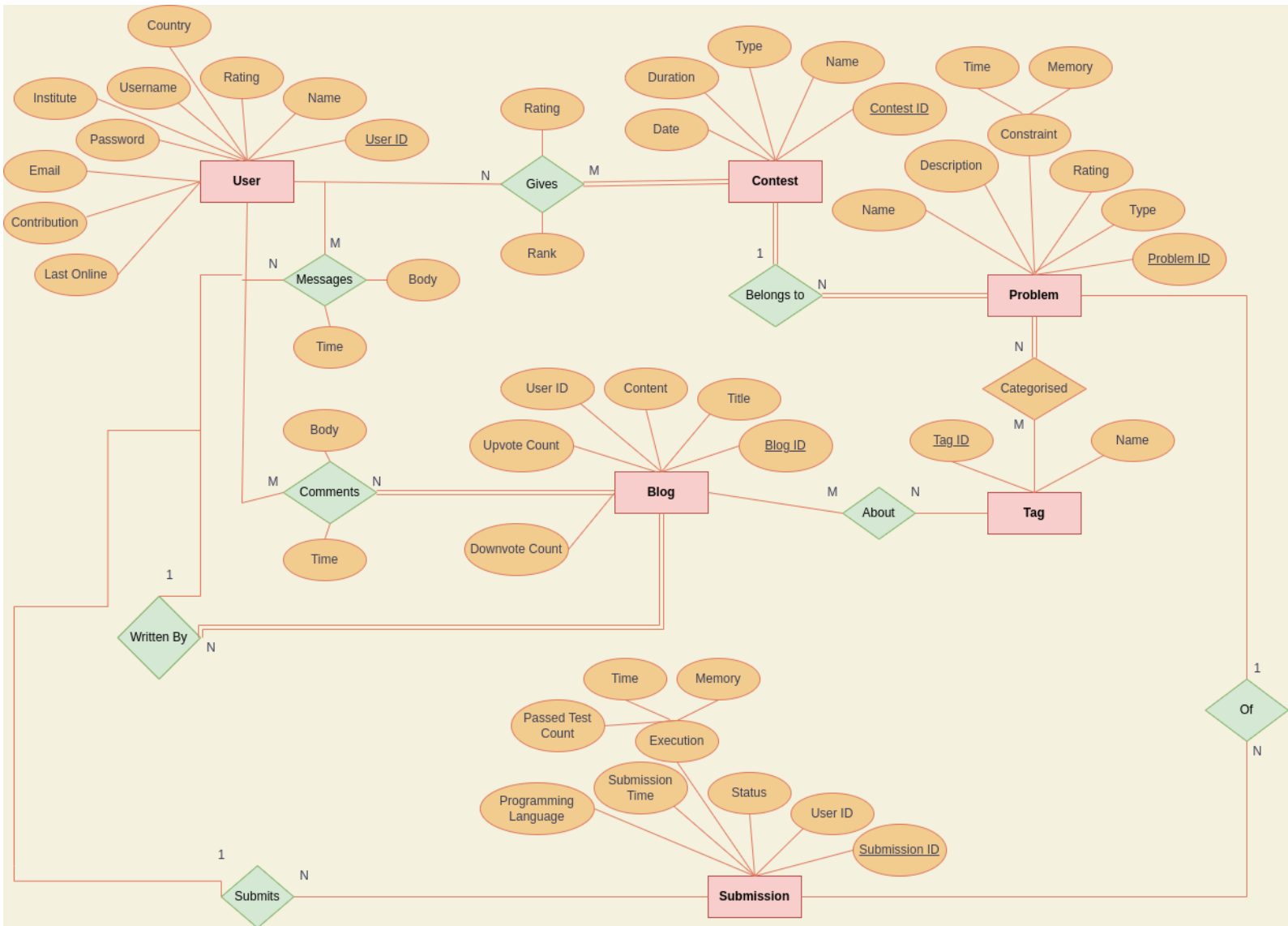
Mini Project Final Report

Team -

Shashank Sundar (PES1UG21CS558)

Shivam Goel (PES1UG21CS561)

ER Diagram



Query Executions and Screenshots

1. Initialize

The screenshot shows the 'Competitive Programming Platform' interface with the 'Initialize' operation selected. The 'Database Creation' section displays MySQL connection information and a command to connect to the server. The 'Data Initialization' section shows a 'Create' button and a message 'User table created successfully'.

Options

Operation: Initialize

Competitive Programming Platform

Database Creation

MySQL Connection Information

```
{
  "SQL_USER": "competitive-programming-dbms-admin",
  "SQL_PASSWORD": "5up3r-s3cur3-p4ssw0rd",
  "SQL_HOST": "localhost",
  "SQL_PORT": "3306"
}
```

Connect to server with using the following command:

```
mysql -u competitive-programming-dbms-admin -p'5up3r-s3cur3-p4ssw0rd' -h localhost -P 3306
```

Database Name: cpdbms_demo

Create

Data Initialization

Create all tables by executing all commands or individually executing

Execute All

User

See query

Execute

User table created successfully

Contest

See query

```
create table if not exists contest (
  contest_id int primary key,
  name varchar(50) not null,
  type varchar(10) not null,
  duration int not null default 60,
  start_time timestamp
```

2. Create

The screenshot shows the 'Competitive Programming Platform' interface with the 'Create' operation selected. The 'Initialize With Random Data' section displays a table of data for various entities. The 'Generate Random Data' button is highlighted, and a message 'Data inserted into table "user" successfully' is shown.

Options

Operation: Create

Competitive Programming Platform

Initialize With Random Data

User	Problem
100	50
Contest	Categorized
10	45
Blog	Message
20	300
Tag	Submission
36	500
Comment	Gives
50	600
About	
20	

Generate Random Data

View Data

Insert Data

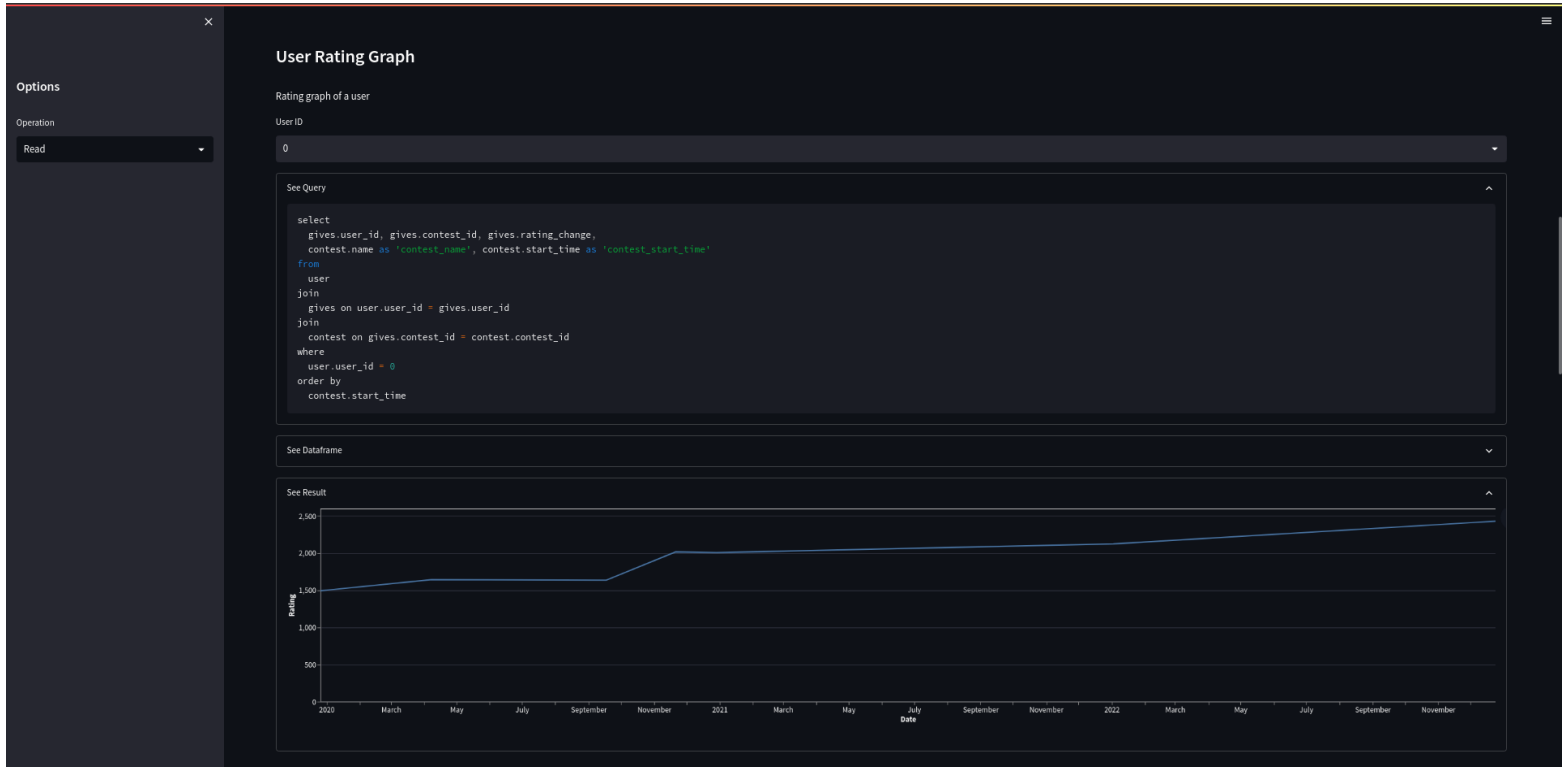
Data inserted into table "user" successfully

Data inserted into table "contest" successfully

Data inserted into table "blog" successfully

Data inserted into table "tag" successfully

3. Read



4. Update

Options

Operation

Update

Competitive Programming Platform

Select table

Table

blog

Selection Attribute

blog_id

Blog ID

0

Change Attribute

content

Content

This is the updated content of the blog. You can change any attribute! This demonstrates changing the 'content' attribute of 'blog'.

132/1000

```
update
  blog
set
  content = "This is the updated content of the blog. You can change any attribute! This demonstrates changing the 'content' attribute of 'blog'."
where
  blog_id = 0
```

Update

Update executed successfully

5. Delete

Options

Operation

Delete

Competitive Programming Platform

Select table

Table

blog

Attribute

user_id

User ID

1

```
delete from
  blog
where
  user_id = 1
```

Delete

Delete executed successfully

Made with Streamlit

6. Custom Query

Options

Operation

Query

Competitive Programming Platform

Custom Query

Enter custom query

select * from blog where blog_id = 0;

Execute Custom Query

Executing query:

```
select * from blog where blog_id = 0;
```

Query Results

	blog_id	title	content	upvote_count	downvote_count	user_id	time	
	0	0	molestiae voluptas soluta	This is the updated content of the blog. You can change any attribute! This demonstrates changing the `content`	414	71	9	2023-01-28T01:08:22

Made with Streamlit

7. Cleanup

