Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Project Title: Programming Tracker

Group members detail: Keval Behera 92100133039

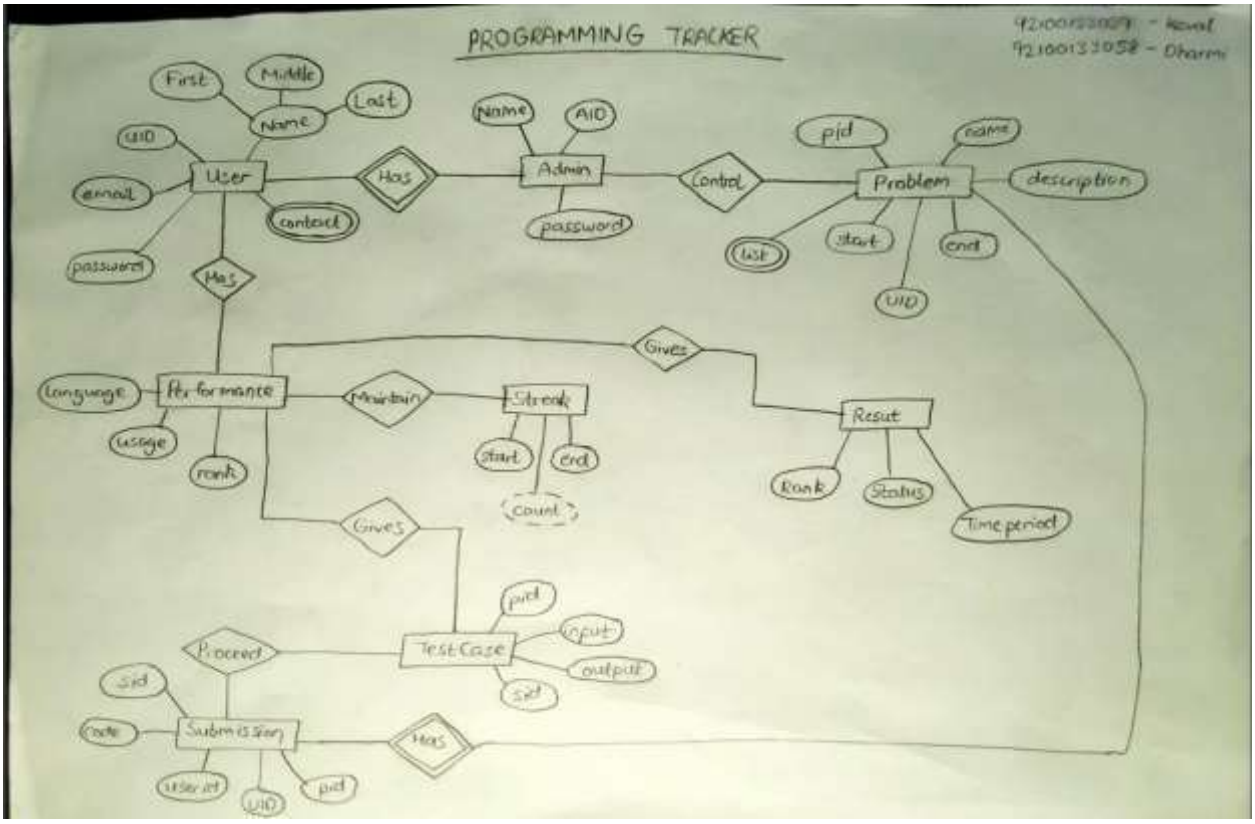
Dharmi Javiya 92100133058

Abstract:

In our project we are applying the logic we learn from DBMS and IWT. Our Problem Statement based on our competitive programming, in that they are giving links every day so on the basis of it we will be making a website in which there are elements like Login/Sign up, Levels such as Easy,Moderate,Difficult. And in that we are planning to add a compiler. From the database, we will fetch the details of the user and record their performance.

There will be extra features like the maintaining of daily programming streak, overall leaderboard across different programming languages. If they continue their programming and maintain their streak for a month then we are going to post their performance chart on linkedIn by using our official account of PROGRAMMING TRACKER.

ER Diagram:



Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

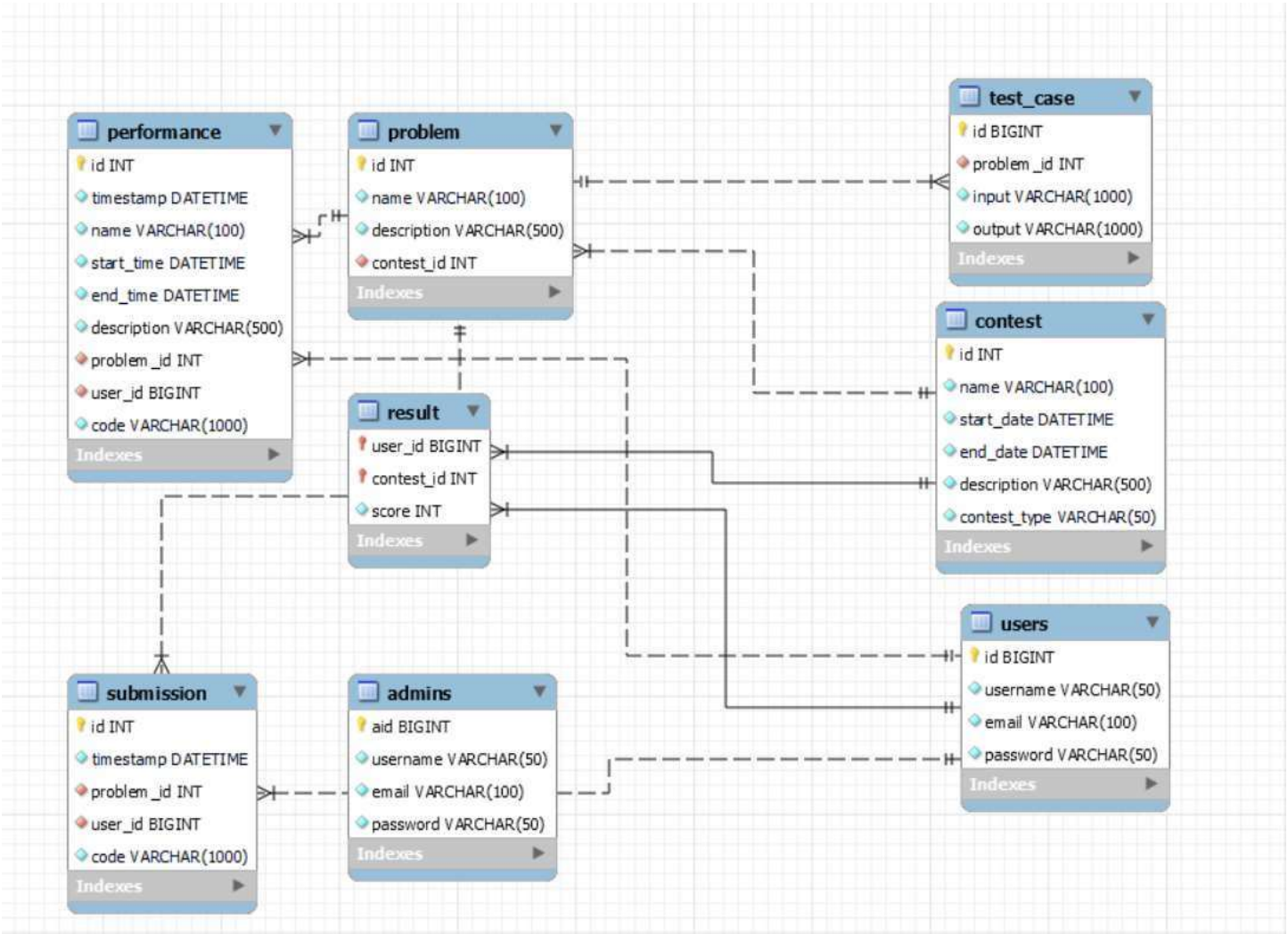


Table schema: User-

id SERIAL PRIMARY KEY,

username VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL,

password VARCHAR(50) NOT NULL ,

datejoin TIMESTAMP DEFAULT

NOW();

Admin - aid SERIAL PRIMARY KEY,

username VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL,

password VARCHAR(50) NOT NULL

Contest- id INT PRIMARY KEY, name

VARCHAR(100) NOT NULL, start_date

DATETIME NOT NULL, end_date

DATETIME NOT NULL, description

VARCHAR(500) NOT NULL,

contest_type VARCHAR(50) NOT NULL

Problem - id INT PRIMARY KEY, name

VARCHAR(100) NOT NULL, description

VARCHAR(500) NOT NULL, contest_id

INT NOT NULL,

FOREIGN KEY (contest_id) REFERENCES Contest(id) Submission - id INT PRIMARY KEY, timestamp DATETIME NOT NULL, problem_id INT NOT NULL, user_id SERIAL NOT NULL, code VARCHAR(1000) NOT NULL,

FOREIGN KEY (problem_id) REFERENCES Problem(id),

FOREIGN KEY (user_id) REFERENCES users(id)

Performance - id INT PRIMARY KEY,

timestamp DATETIME NOT NULL,

name VARCHAR(100) NOT NULL,

start_time DATETIME NOT NULL,

end_time DATETIME NOT NULL,

description VARCHAR(500) NOT NULL,

problem_id INT NOT NULL, user_id

SERIAL NOT NULL, code

VARCHAR(1000) NOT NULL,

FOREIGN KEY (problem_id) REFERENCES Problem(id),

FOREIGN KEY (user_id) REFERENCES users(id)

Test_Case - id SERIAL PRIMARY

KEY, problem_id INT NOT NULL,

input VARCHAR(1000) NOT NULL,

output VARCHAR(1000) NOT NULL,

FOREIGN KEY (problem_id) REFERENCES Problem(id)

Result -

user_id SERIAL NOT NULL,

contest_id INT NOT NULL,

PRIMARY KEY (user_id, contest_id),

FOREIGN KEY (user_id) REFERENCES users(id),

FOREIGN KEY (contest_id) REFERENCES Contest(id)

Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

SQL Queries:

User:

```
CREATE TABLE users(
  id SERIAL PRIMARY KEY,
  username VARCHAR(50) NOT NULL,
  email VARCHAR(100) NOT NULL,
  password VARCHAR(50) NOT NULL
);

INSERT INTO users (username, email, password)
VALUES
('johndoe', 'johndoe@example.com', 'password123'),
('janedoe', 'janedoe@example.com', 'password456'),
('bobsmith', 'bobsmith@example.com', 'password789'),
('amandajones', 'amandajones@example.com', 'password101'),
('davidsmith', 'davidsmith@example.com', 'password202');

SELECT * FROM users;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	username	email	password
1	johndoe	johndoe@example.com	password123
2	janedoe	janedoe@example.com	password456
3	bobsmith	bobsmith@example.com	password789
4	amandajones	amandajones@example.com	password101
5	davidsmith	davidsmith@example.com	password202
NULL	NULL	NULL	NULL

Admin –

```
CREATE TABLE admins (
    aid SERIAL PRIMARY KEY,
    username VARCHAR(50) NOT NULL,
    email VARCHAR(100) NOT NULL,
    password VARCHAR(50) NOT NULL
);

INSERT INTO admins (username, email, password)
VALUES
('admin1', 'admin1@example.com', 'password123'),
('admin2', 'admin2@example.com', 'password456'),
('admin3', 'admin3@example.com', 'password789'),
('admin4', 'admin4@example.com', 'password101'),
('admin5', 'admin5@example.com', 'password202');

SELECT * FROM admins;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

aid	username	email	password
1	admin1	admin1@example.com	password123
2	admin2	admin2@example.com	password456
3	admin3	admin3@example.com	password789
4	admin4	admin4@example.com	password101
5	admin5	admin5@example.com	password202
NULL	NULL	NULL	NULL

Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Contest-

```
CREATE TABLE Contest (
    id INT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    start_date DATETIME NOT NULL,
    end_date DATETIME NOT NULL,
    description VARCHAR(500) NOT NULL,
    contest_type VARCHAR(50) NOT NULL
);
INSERT INTO Contest (id, name, start_date, end_date, description, contest_type)
VALUES
(1, 'CodeMaster 2023', '2023-05-01 09:00:00', '2023-05-02 17:00:00', 'A programming contest for college students', 'Individual'),
(2, 'HackTheWorld 2023', '2023-08-10 10:00:00', '2023-08-11 18:00:00', 'A global programming contest for professionals', 'Team'),
(3, 'JuniorCoder 2024', '2024-02-15 08:00:00', '2024-02-16 16:00:00', 'A programming contest for high school students', 'Individual'),
(4, 'BuildIt 2024', '2024-06-20 11:00:00', '2024-06-21 19:00:00', 'A programming contest for software developers', 'Team'),
(5, 'CodeFiesta 2025', '2025-09-05 07:00:00', '2025-09-06 15:00:00', 'A programming contest for beginners', 'Individual');

SELECT * FROM Contest;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	name	start_date	end_date	description	contest_type
1	CodeMaster 2023	2023-05-01 09:00:00	2023-05-02 17:00:00	A programming contest for college students	Individual
2	HackTheWorld 2023	2023-08-10 10:00:00	2023-08-11 18:00:00	A global programming contest for professionals	Team
3	JuniorCoder 2024	2024-02-15 08:00:00	2024-02-16 16:00:00	A programming contest for high school students	Individual
4	BuildIt 2024	2024-06-20 11:00:00	2024-06-21 19:00:00	A programming contest for software developers	Team
5	CodeFiesta 2025	2025-09-05 07:00:00	2025-09-06 15:00:00	A programming contest for beginners	Individual

Problems-

```
CREATE TABLE Problem (
    id INT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    description VARCHAR(500) NOT NULL,
    contest_id INT NOT NULL,
    FOREIGN KEY (contest_id) REFERENCES Contest(id)
);
INSERT INTO Problem (id, name, description, contest_id)
VALUES
(1, 'Problem A', 'Write a program to find the maximum element in an array.', 1),
(2, 'Problem B', 'Write a program to count the number of vowels in a string.', 1),
(3, 'Problem C', 'Write a program to calculate the factorial of a number.', 1),
(4, 'Problem D', 'Write a program to find the shortest path between two nodes in a graph.', 2),
(5, 'Problem G', 'Write a program to find the sum of all prime numbers up to a given limit.', 3),
(6, 'Problem H', 'Write a program to find the GCD of two numbers.', 3),
(7, 'Problem I', 'Write a program to implement a bubble sort algorithm.', 3);
SELECT * FROM Problem;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	name	description	contest_id
1	Problem A	Write a program to find the maximum element in...	1
2	Problem B	Write a program to count the number of vowels...	1
3	Problem C	Write a program to calculate the factorial of a n...	1
4	Problem D	Write a program to find the shortest path betw...	2
5	Problem G	Write a program to find the sum of all prime nu...	3
6	Problem H	Write a program to find the GCD of two numbers.	3

Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Submission -

```
46 • CREATE TABLE submission (
47     id INT PRIMARY KEY,
48     timestamp DATETIME NOT NULL,
49     problem_id INT NOT NULL,
50     users_id SERIAL NOT NULL,
51     code VARCHAR(1000) NOT NULL,
52     FOREIGN KEY (problem_id) REFERENCES problem(id),
53     FOREIGN KEY (users_id) REFERENCES users(id)
54 );
55
56 • INSERT INTO submission (id, timestamp, problem_id, users_id, code)
57   VALUES (1, '2023-04-20 10:30:00', 1, 1, 'Example code for problem 1 by user 1');
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id	timestamp	problem_id	users_id	code
1	2023-04-20 10:30:00	1	1	Example code for problem 1 by user 1

Performance:

```
62 • CREATE TABLE performance (
63     timestamp DATETIME NOT NULL,
64     name VARCHAR(100) NOT NULL,
65     start_time TIMESTAMP NOT NULL,
66     end_time TIMESTAMP NOT NULL,
67     description VARCHAR(500) NOT NULL,
68     problem_id INT NOT NULL,
69     user_id SERIAL NOT NULL,
70     code VARCHAR(1000) NOT NULL,
71     FOREIGN KEY (problem_id) REFERENCES problem(id),
72     FOREIGN KEY (user_id) REFERENCES users(id)
73 );
74 • INSERT INTO performance (timestamp, name, start_time, end_time, description, problem_id, user_id, code)
75   VALUES ('2023-04-20 13:30:00', 'Good', '2023-04-20 14:00:00', '2023-04-20 15:30:00', 'Description of Performance 1', 1, 1, 'Example code for Performance 1 by User 1');
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

timestamp	name	start_time	end_time	description	problem_id	user_id	code
2023-04-...	Good	2023-04-20 14:00:00	2023-04-20 15:30:00	Description of Performance 1	1	1	Example code for Performance 1 by User 1

Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Test Case:

```
77 • CREATE TABLE test_case (
78     id INT PRIMARY KEY,
79     problem_id INT NOT NULL,
80     input VARCHAR(1000) NOT NULL,
81     output VARCHAR(1000) NOT NULL,
82     FOREIGN KEY (problem_id) REFERENCES problem(id)
83 );
84 • INSERT INTO test_case (id, problem_id, input, output)
85   VALUES (1, 1, 'Input for test case 1 of problem 1', 'Output for test case 1 of problem 1');
86 • SELECT * FROM test_case;
87
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id	problem_id	input	output
1	1	Input for test case 1 of problem 1	Output for test case 1 of problem 1
NULL	NULL	NULL	NULL

Result:

```
112 •     CREATE TABLE Result (
113         user_id SERIAL NOT NULL,
114         contest_id INT NOT NULL,
115         score INT NOT NULL,
116         PRIMARY KEY (user_id, contest_id),
117         FOREIGN KEY (user_id) REFERENCES users(id),
118         FOREIGN KEY (contest_id) REFERENCES Contest(id)
119       );
120 • INSERT INTO Result (user_id, contest_id, score) VALUES (2, 1, 75);
121 • SELECT * FROM Result;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

user_id	contest_id	score
2	1	75
NULL	NULL	NULL

Nested Query - Retrieve all submissions for a specific problem by a specific user:

```
112 •  SELECT *
113    FROM submission
114    WHERE problem_id = (
115      SELECT id
116      FROM problem
117      WHERE name = 'Problem 1'
118    ) AND users_id = (
119      SELECT id
120      FROM users
121      WHERE username = 'Rahul Sharma'
122    );
123
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id	timestamp	problem_id	users_id	code
1	2023-04-20 10:30:00	1	1	Example code for problem 1 by user 1
NULL	NULL	NULL	NULL	NULL

Nested Query - Retrieve all submissions for a specific contest:

```
136 •  SELECT *
137    FROM submission
138    WHERE problem_id IN (
139      SELECT id
140      FROM problem
141      WHERE contest_id = (
142        SELECT id
143        FROM contest
144        WHERE id = 1
145      )
146    );
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id	timestamp	problem_id	users_id	code
1	2023-04-20 10:30:00	1	1	Example code for problem 1 by user 1
NULL	NULL	NULL	NULL	NULL

Marwadi University Marwadi Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Nested Query - Retrieve all submissions for a specific contest along with the corresponding problem and user information:

```
185  SELECT s.*, p.name AS problem_name, u.username AS user_username
186  FROM submission s
187  JOIN problem p ON s.problem_id = p.id
188  JOIN users u ON s.users_id = u.id
189  WHERE p.contest_id = (
190    SELECT id
191    FROM contest
192    WHERE id = 1
193  );
194
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	timestamp	problem_id	users_id	code	problem_name	user_username
1	2023-04-20 10:30:00	1	1	Example code for problem 1 by user 1	Problem 1	Rahul Sharma

Aggregated Query - Retrieve the average score of all users in a specific contest:

```
107
108  SELECT AVG(score) AS avg_score
109  FROM result
110  WHERE contest_id = 1;
111
```

Result Grid | Filter Rows:

avg_score
85.0000

Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

Aggregated Query - Retrieve the count of submissions made by each user:

```
131
132 •  SELECT users_id, COUNT(*) AS submission_count
133    FROM submission
134    GROUP BY users_id;
135
```

Result Grid | Filter Rows: | Export: | Wrap

users_id	submission_count
1	1

Join Query - fetches the usernames and scores of all users who participated in the Contest:

```
123 •  SELECT u.username, r.score
124    FROM users u
125    JOIN Result r ON u.id = r.user_id
126    WHERE r.contest_id = 1
127
128
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

username	score
janedoe	75

Join Query – fetches users and respective scores for contest_id 1 and rest null are shown:

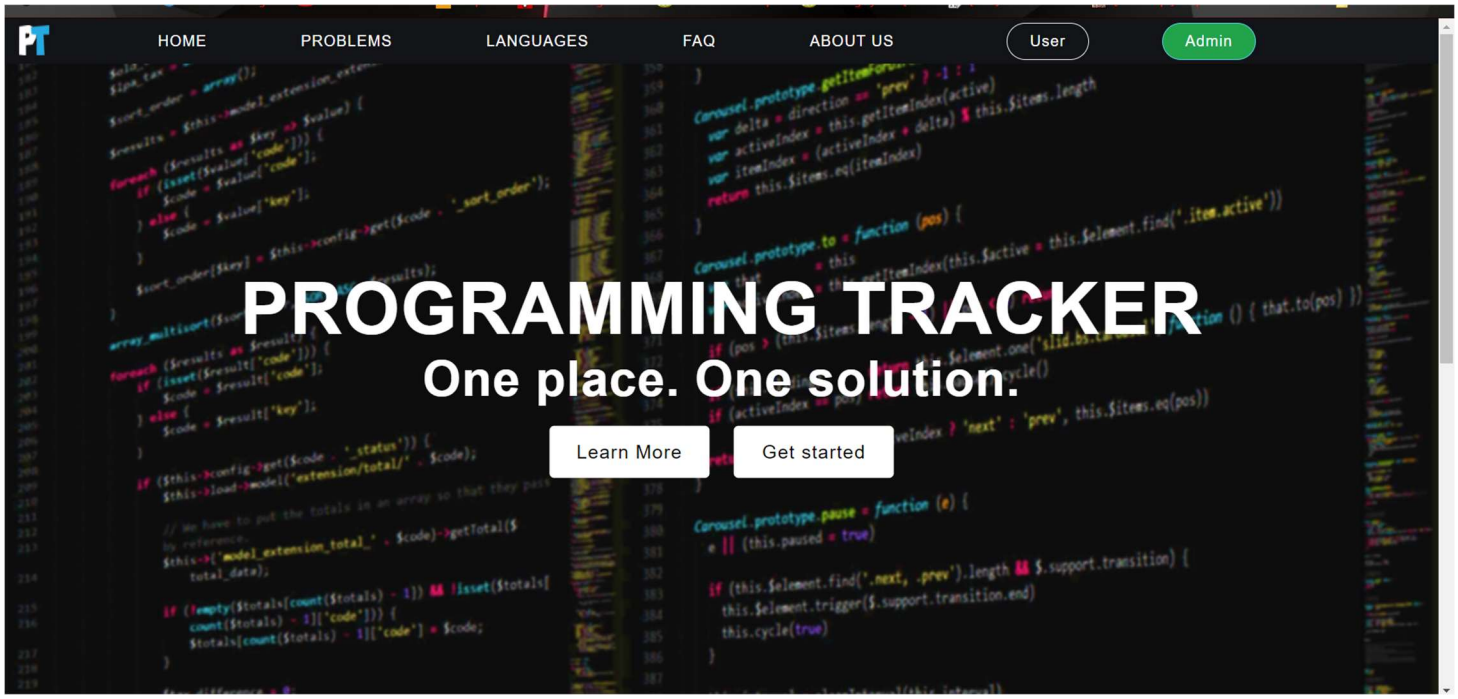
```
134 •  SELECT u.username, r.score
135    FROM users u
136    LEFT JOIN Result r ON u.id = r.user_id
137    WHERE r.contest_id = 1 OR r.contest_id IS NULL;
138
139
```

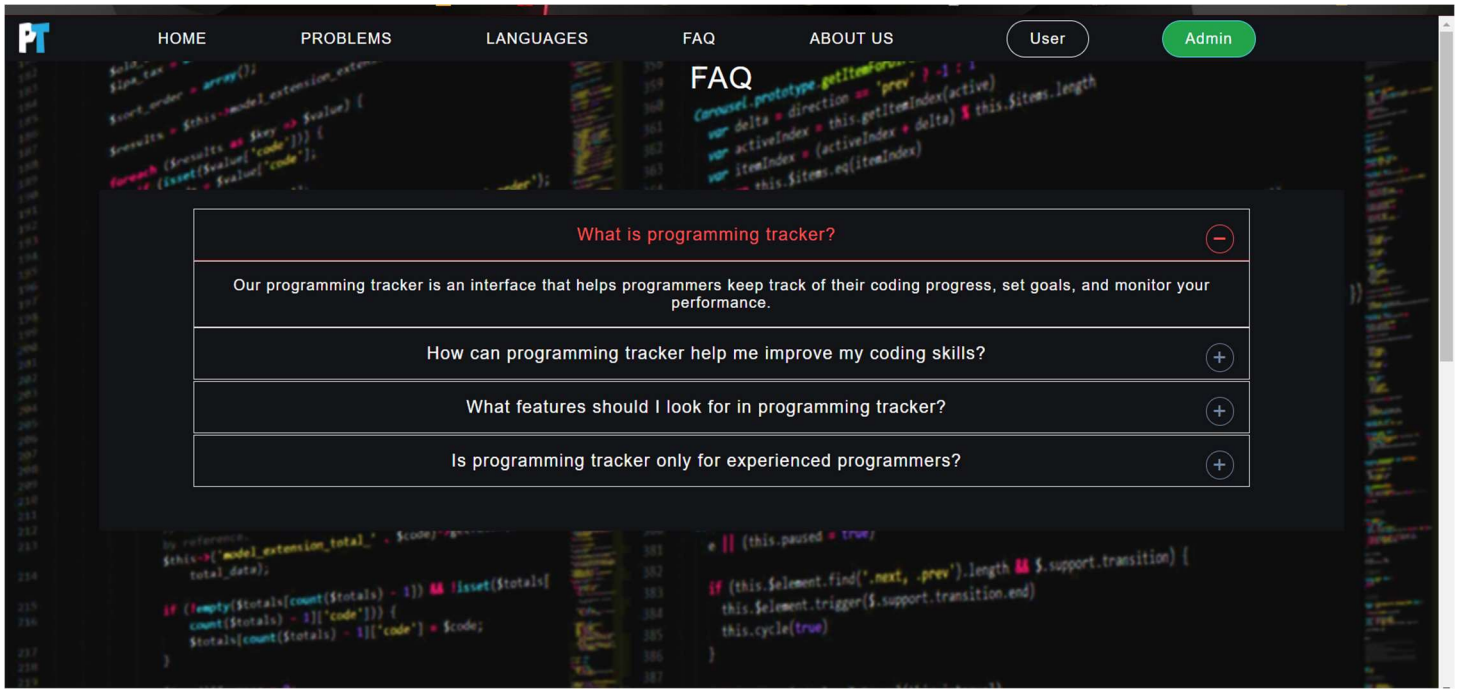
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

username	score
johndoe	NULL
janedoe	75
bobsmith	NULL
amandajones	NULL
davidsmith	NULL

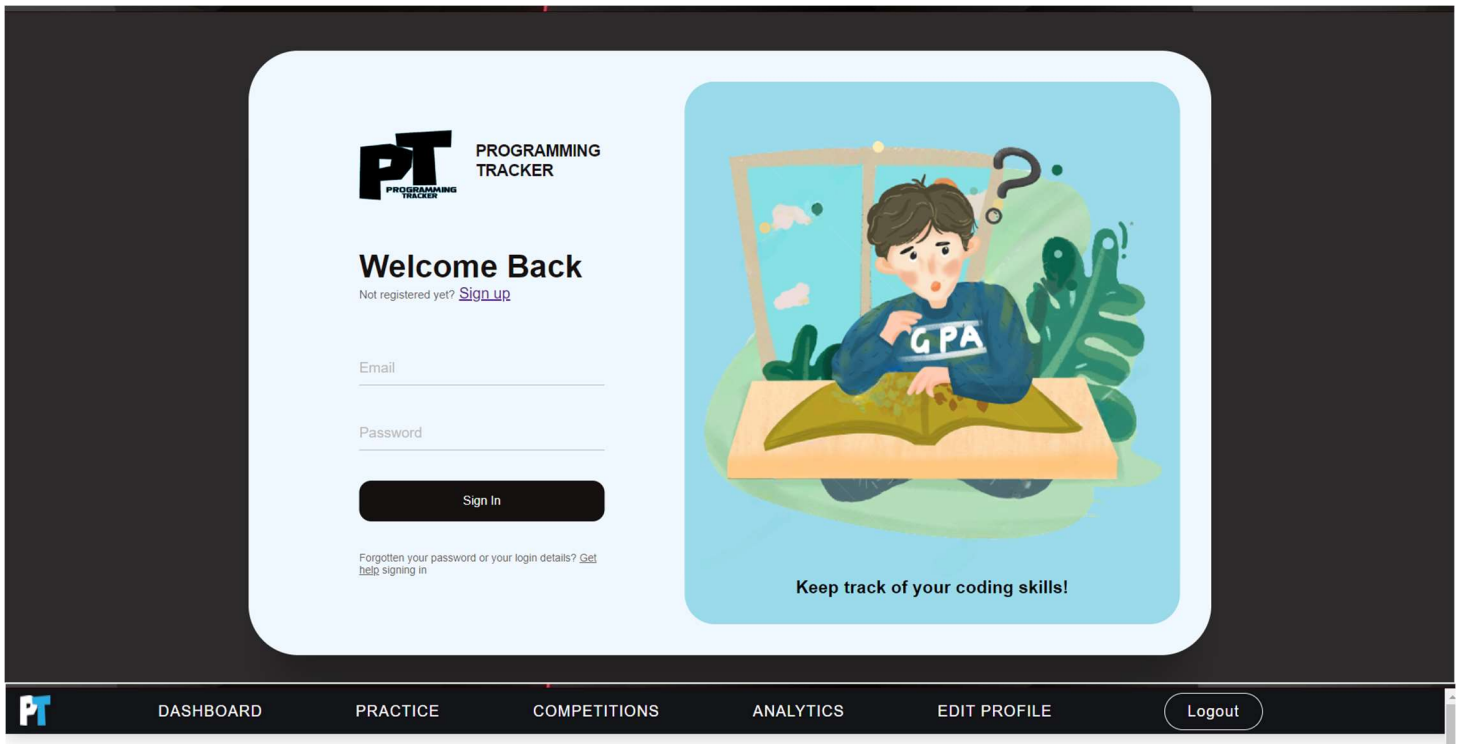
Marwadi University Faculty of Technology Department of Information and Communication Technology	
DBMS + IWT	**PROJECT REPORT**

Screenshots:

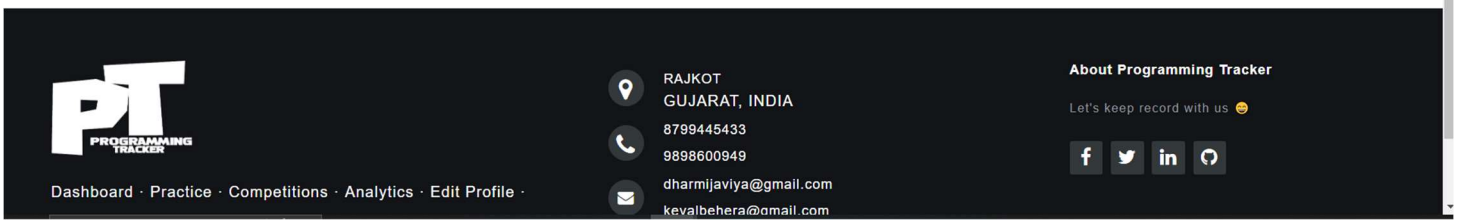




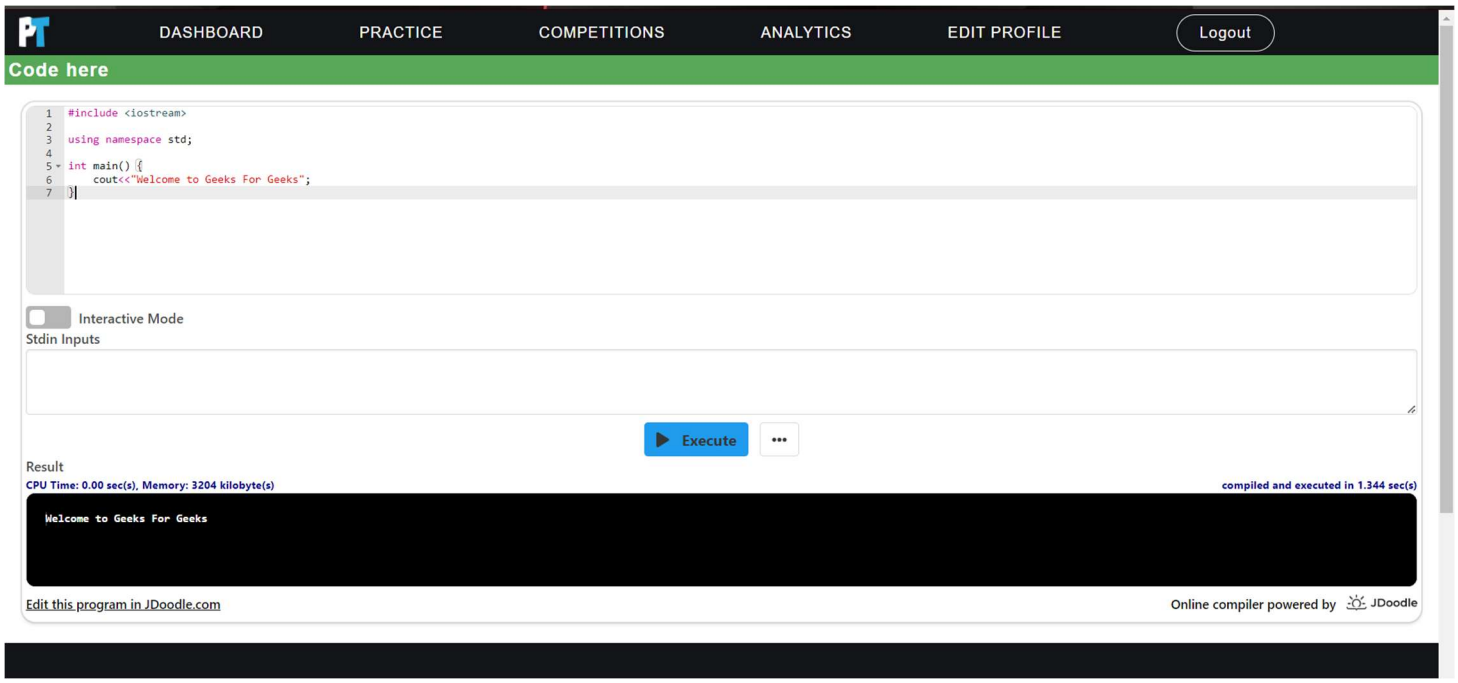
Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**

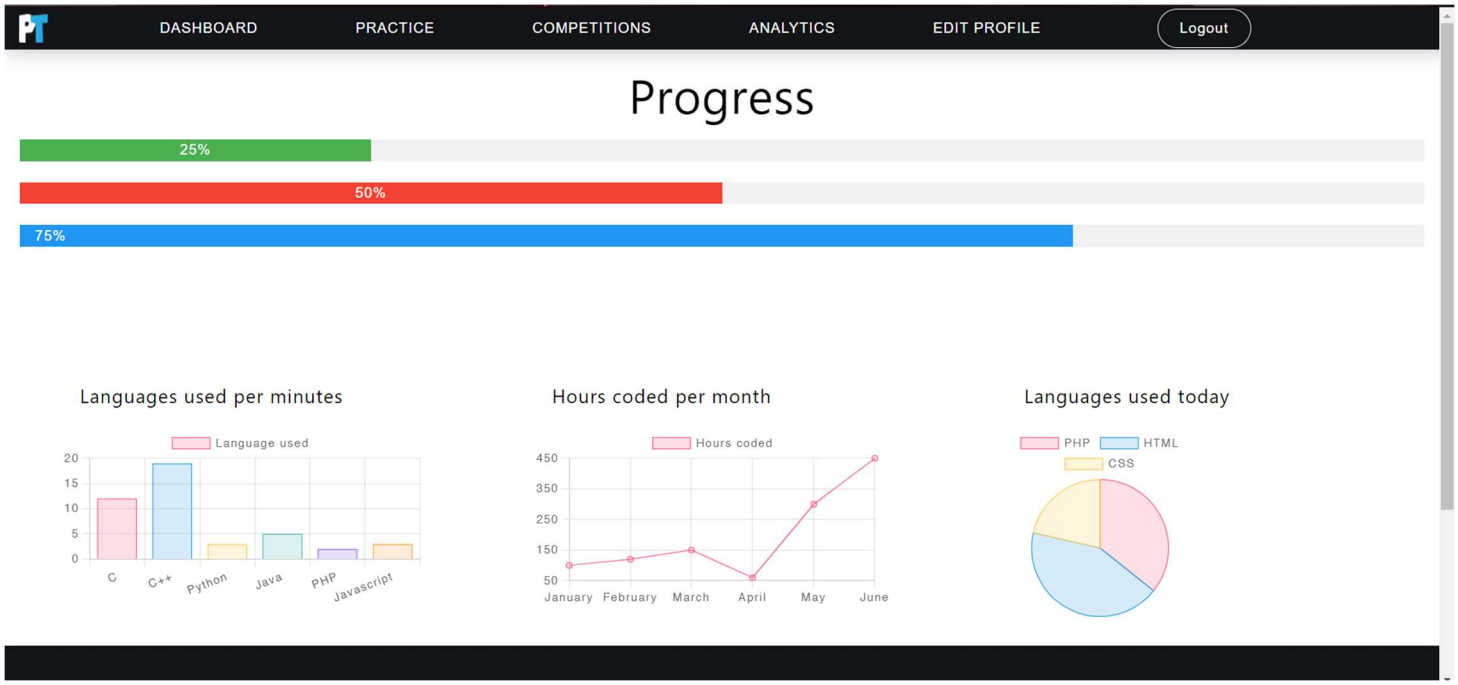


Username:	Keval
Email:	kevalbehera@gmail.com
Joined:	2023-05-09 16:24:53

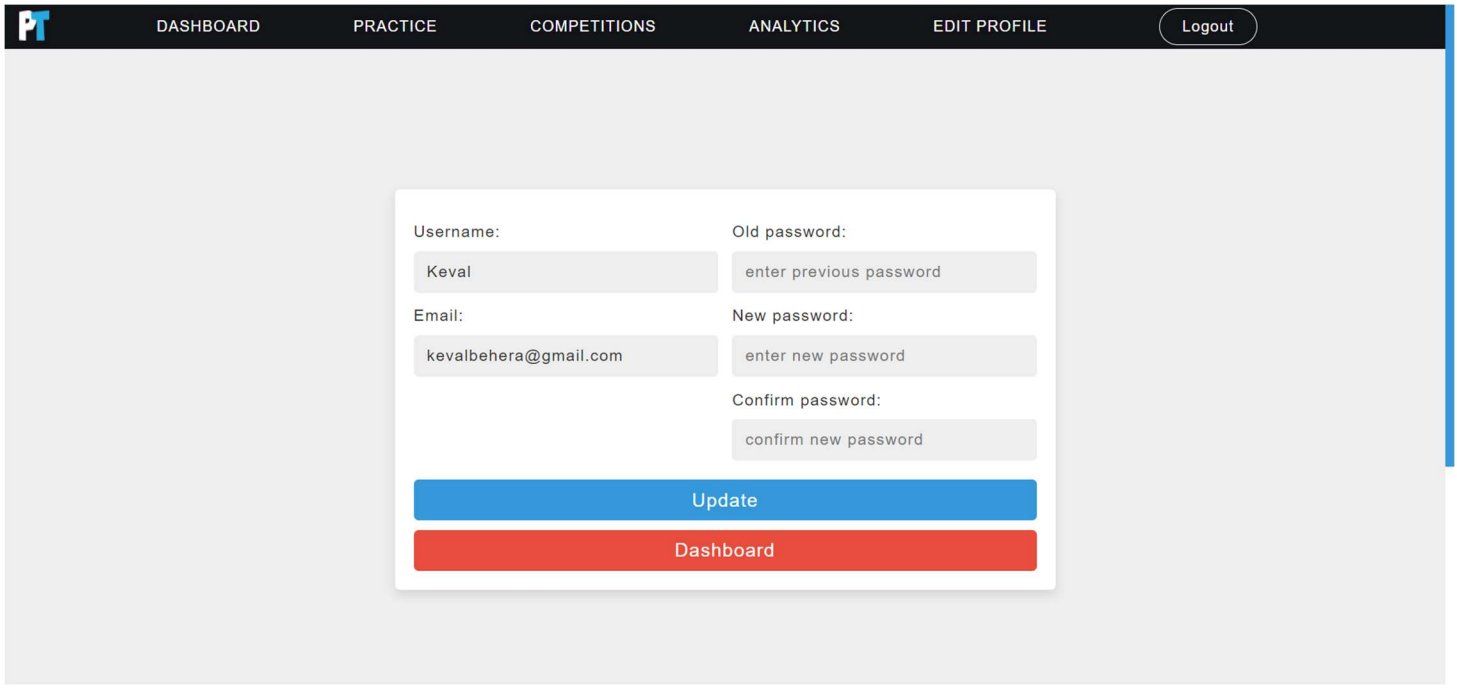


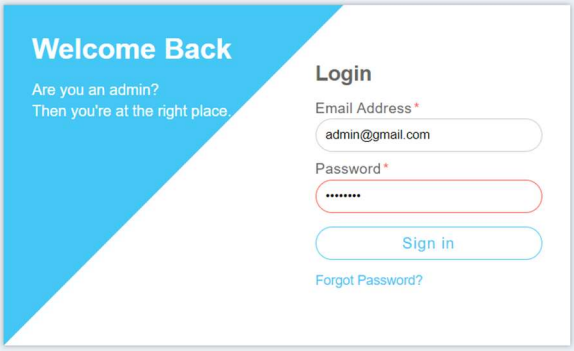
Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**



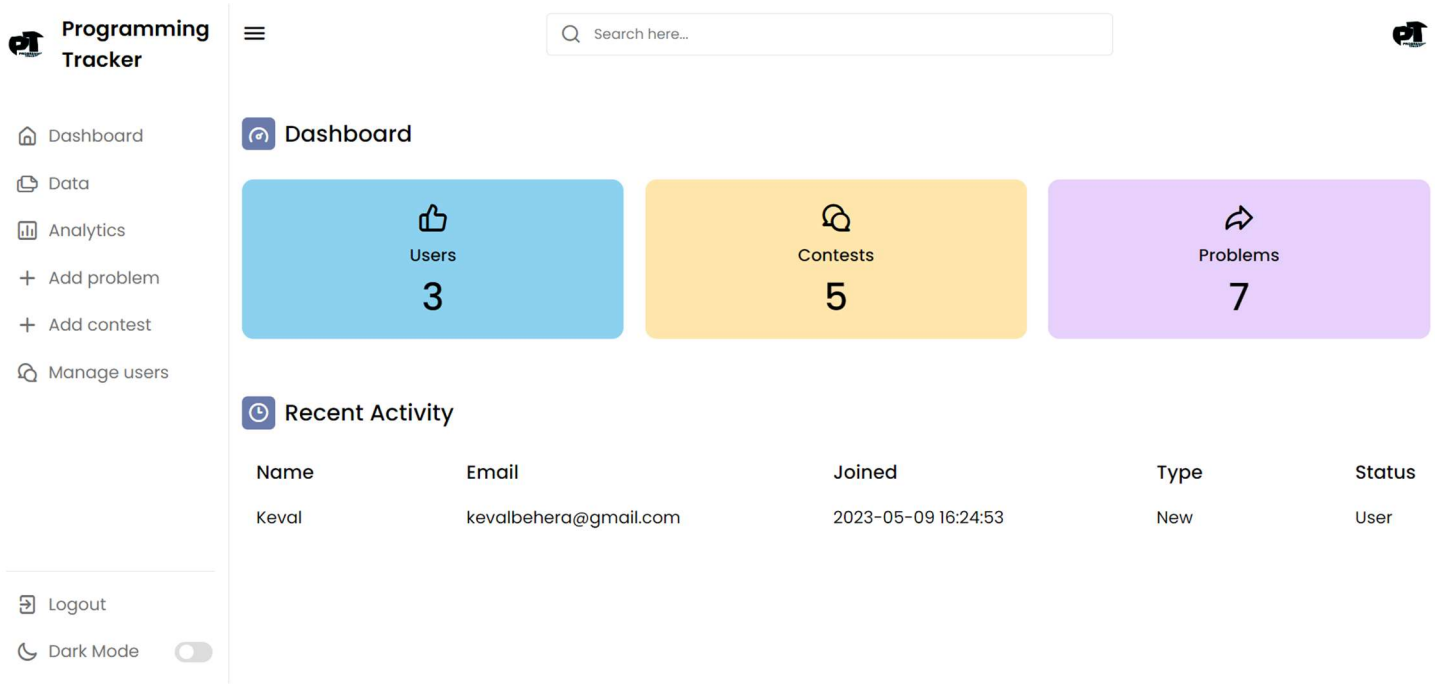


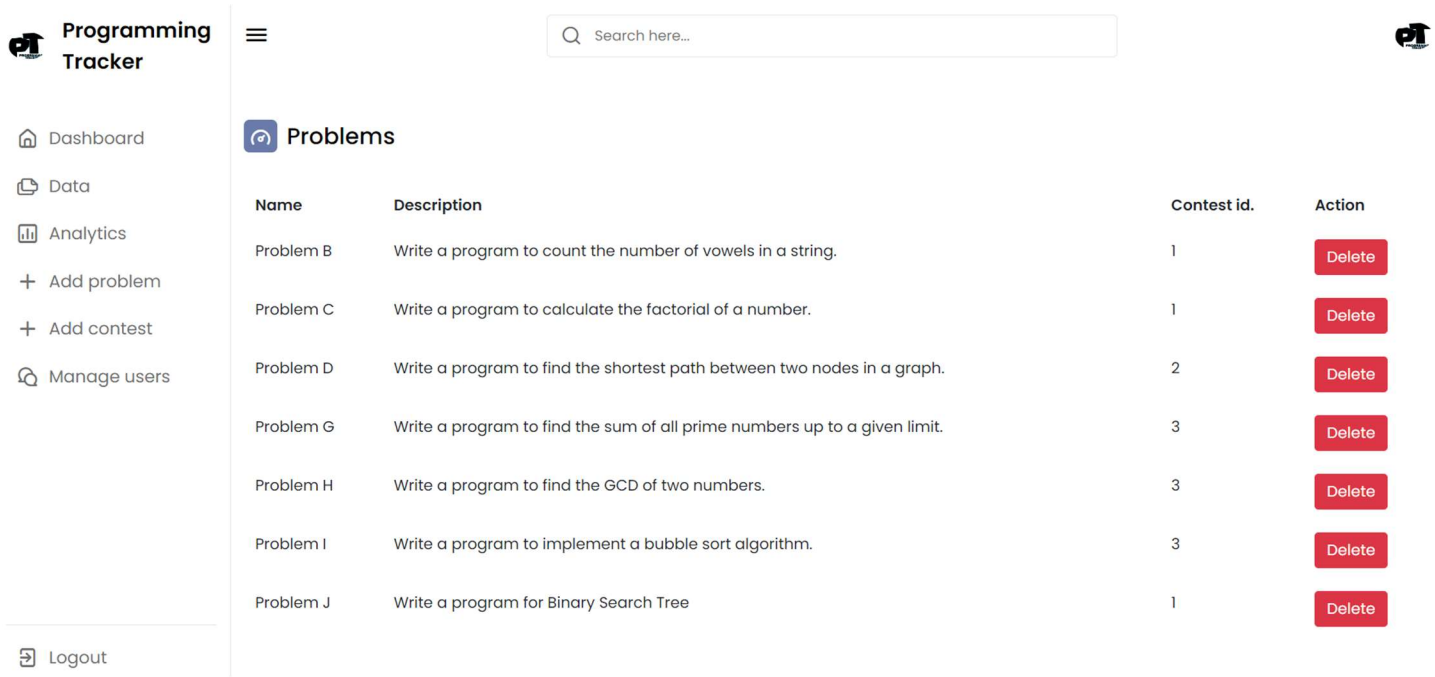
Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**



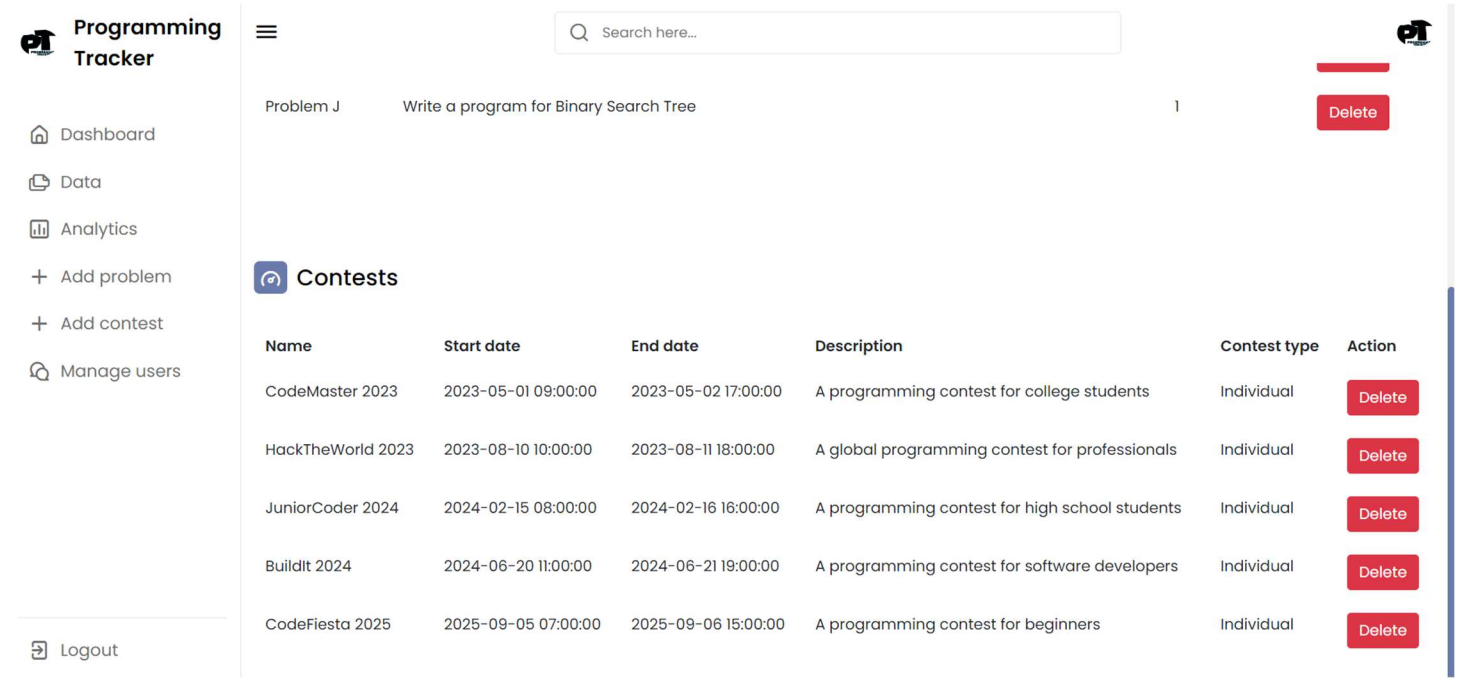


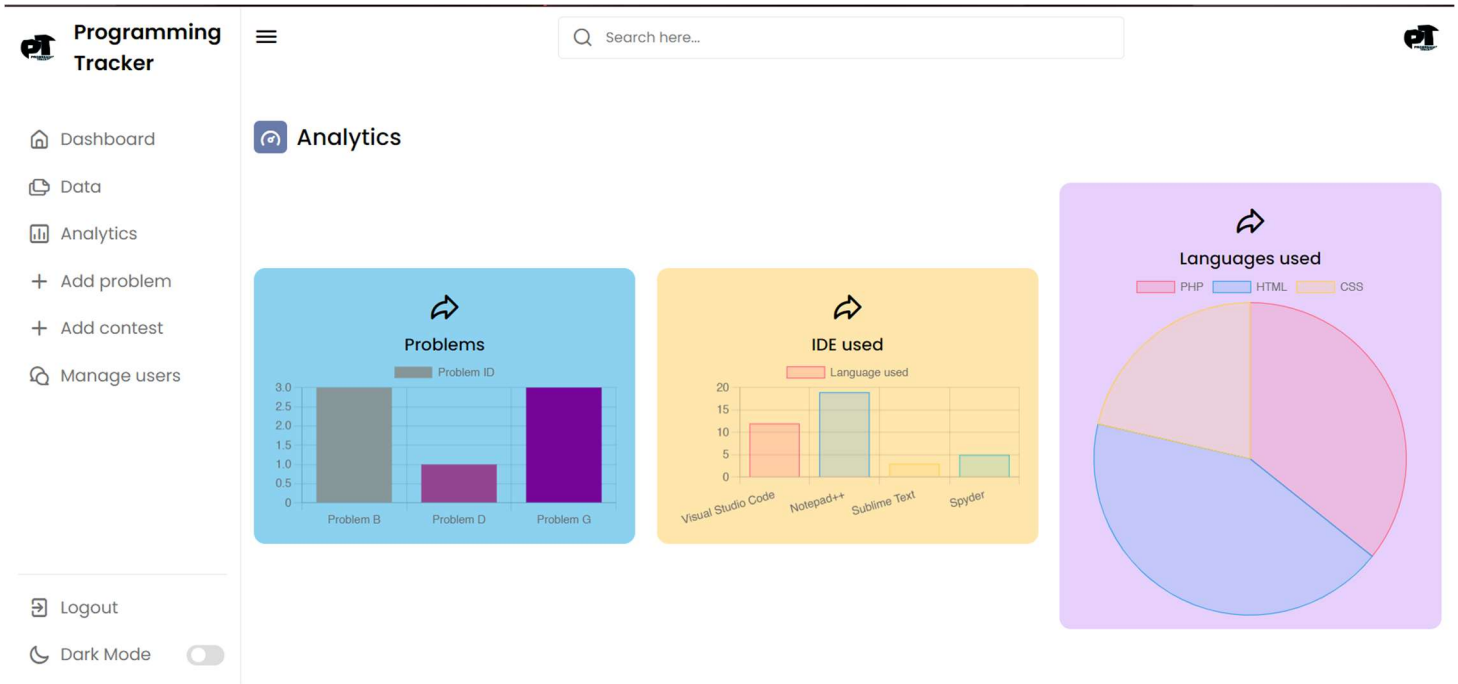
Marwadi University	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**



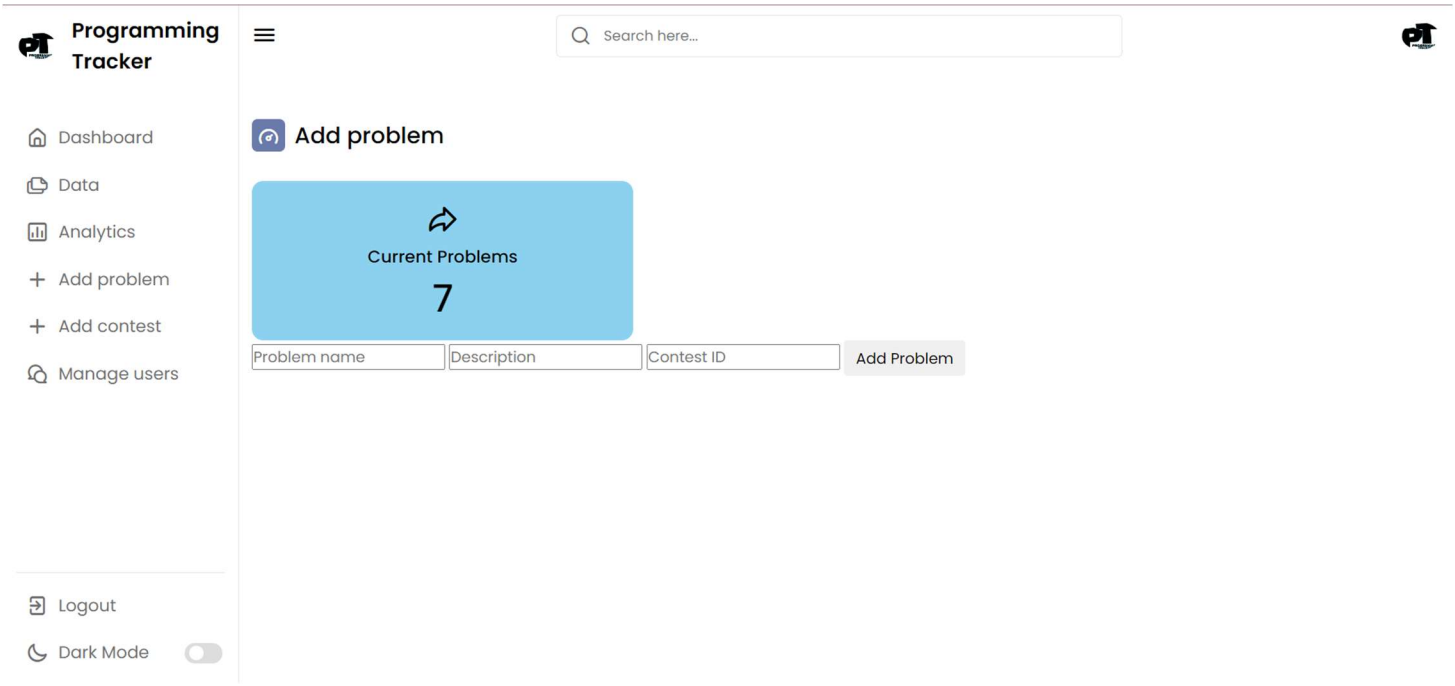


Marwadi University Faculty of Technology Department of Information and Communication Technology	
DBMS + IWT	**PROJECT REPORT**





Marwadi University	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**



Marwadi University Marwad Chandarana Group	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**
DBMS + IWT	**PROJECT REPORT**