



Project Report

On

Contest Hosting Portal



Submitted To

Dr. Vinod Kumar Jain

Department of Computer Science and
Engineering

Submitted By

Naman Lal - 2014109

Rohit Rajwani - 2014144

Saurabh Joshi - 2014159

Shubhank Dubey - 2014174

Table Of Contents

Index	Topic	Page no.
1	Introduction	1
2	Need Of Project	1
3	Sections contained in our project	1
4	Technologies Used	2
5	Database design	2
4	Few Snapshots of GUI	4
5	Few Sample dataset	4
6	Conclusion	5
7	References	5

Introduction

Technology is moving faster and changing standard of lives very fast. Machines have now become a part of our lives today. It has transformed manual work to digital work and that too with great accuracy and efficiency.

In our college when any event has to be organized like quiz events and other programming contests, generally printing of papers and other manual arrangements are done which are really a waste in this technically progressing world. Our main **Objective of this project** is to facilitate hosting of contests on a website so that the above discussed process becomes easy and in turn we are helping our environment too by saving a lot of paperwork. The function of the project is to provide a portal to the event organizers in which they can schedule a contest by easily uploading question and answers based on an administrator identity provided to them.

Need of Project

The project sees its need in solving few underlying problems - >

- We are not sure about how many people will take part in any contest so we generally print a lot of papers to ensure there is no shortage but if users get suddenly low then all things go for waste especially during the college technical fest.
- Accuracy with humans is the most variable factor. If a question set contains error in one or more questions, then we can't make any changes to the hard copy.
- Time for evaluating and compilation of results is not deterministic.

But if the above work is done digitally then to edit the problems or any changes, or to generate results and ranks become very easy. We don't have to think how many people will participate because there is no need to generate any special things and everything is automatically done. Just done while conducting IITJEE ,VIT etc. online exams where you just have to submit one correct option.

Our project contains the following sections ->

Registration – Every member whether he is admin of any contest or not registers through the signup form in the home page. The permission to become an administrator will be provided by the database administrator.

Hosting – Hosting section is made for the administrators so that they can host contest for other people. Every member of portal will be given access to this button/page if and only if he is an administrator. He can set problems, schedule and

also edit problems before the contest ends. Even if a problem is found during the live contest he can edit that too.

Participation – All the registered members in the portal can participate in the contests except the administrator of that contest as he will be given the access to edit problems, view leader boards but he himself being a problem setter can't take part in the contest, he will not be given that access.

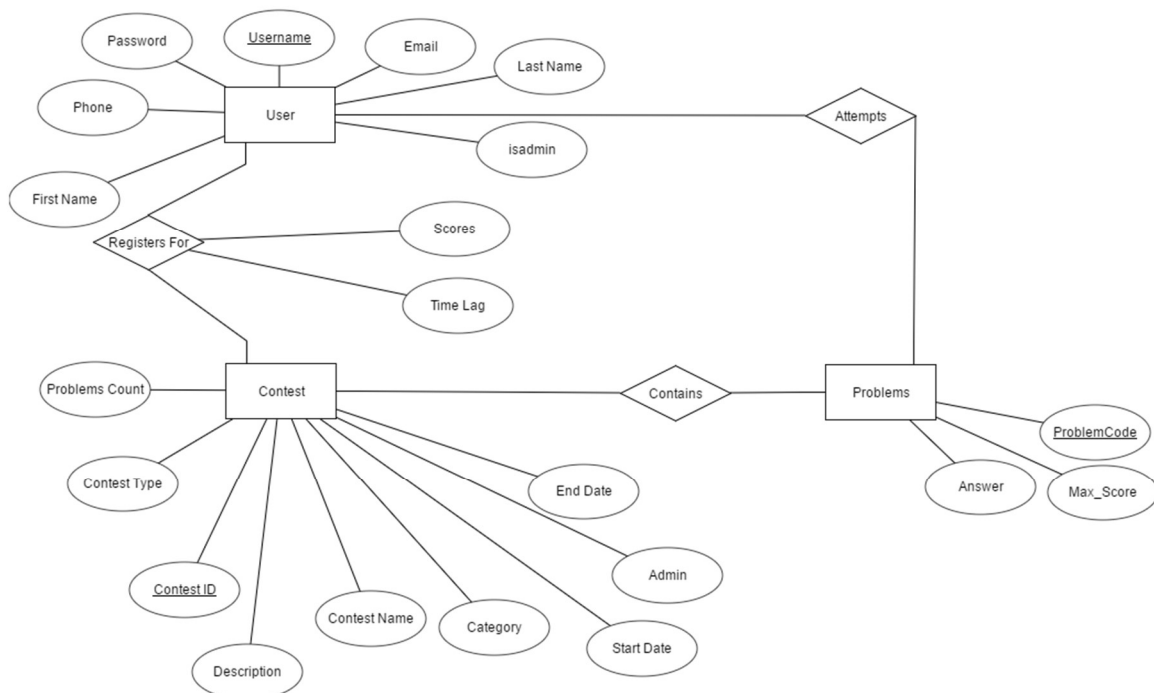
Technologies Used

ER Designing – We have used ERD Plus online tool to design ER MODEL for the database

Front End – For front end we have used Materialize framework (CSS), HTML and JavaScript

Back End – My SQL, PHP

Database Design and ER Model



Our database contains three entities –

1. **User** – This stores details of registered user along with their access privileges as administrators or not. Every data incurred from the registration form is stored here. The attributes of this table are ->
 - **Username** – Uniquely identifies each user in the database. It is primary key of this table
 - Password – Encrypted text used for authentication purpose for logging into the portal
 - Phone – Phone stores phone number of the person which can be used to further contact the person after the contest if he is eligible for some prizes or any other matter
 - Email – Email is just for sending some private notifications or announcements
 - First Name – First Name of the user
 - Last Name – Last Name of the user
 - IsAdmin – This Boolean value is 1 if user is the admin or else it is zero

2. **Contest** – This table stores data of every contest viz. Contest name, start time, end time, Unique identity of each contest, count of problems. The attributes of this table are ->
 - ContestType - Contest Type contains numeric value defining type of contest. Currently we support values 1 and 2 only. 1 for multiple choice questions and 2 for no choice simply solve and enter questions
 - **ContestID** – Unique ID for each contest. It is primary key
 - Description – Description of each event
 - ContestName – Name of the event
 - Category – Category categorizes each contest based on the event type. Currently only three are supported and rest are categorized as others.
 - StartDate – It is an attribute of type Date and Time which stores end time of the contest
 - EndDate – It is an attribute of type Date and Time which stores end time of the contest
 - Rules – Rules store rules and other announcements regarding the project
 - **Admin** - It is foreign key that shows which user is admin

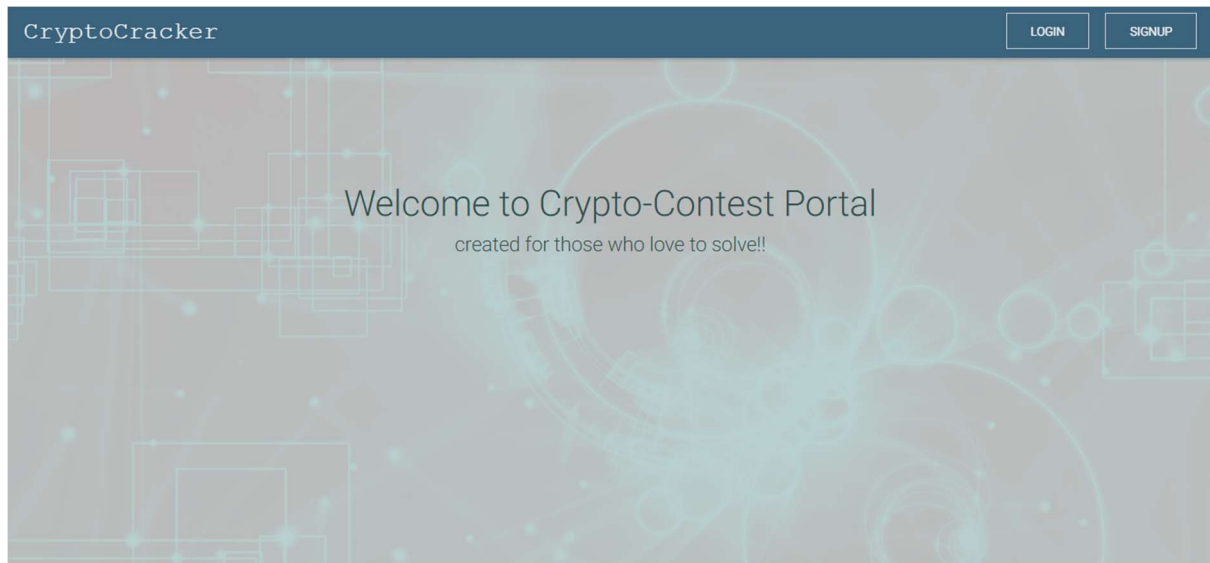
3. **Problems** – This table stores details of problems i.e problem code and its answer along with total points one secures after solving the problem completely. The attributes of this table are - >
 - **Problem Code** – A unique code for each problem and it's the primary key of this table
 - Answer – Answer to that particular problem
 - Max_Score – Max Score is the score one obtains on successfully solving the problem

To Maintain connectivity between the above entity sets we have made few relationship sets which help us in proper retrieval of data. We have three binary relationships –

1. **Registers For** – This relationship contains information about which user has registered for which contest. To manage ranklist in the contest we have used the same table and for doing that we added few more attributes. The attributes of this relationship are –
 - **Username** – Foreign Key Referencing the Username of User table
 - **Contest ID** – Foreign Key Referencing the ContestID of contest table. The above attribute with this combination tracks the user has registered for the contest or not.
 - Scores – The total score by a particular user registered for a contest, it has initial value as 0
 - Time Lag – This stores the total time in seconds taken by the contestant in solving the problems and will be used to break tie between same points people.
 - Rank – Rank stores the rank of the user in that particular contest. This in particular is not required but for optimization we have used it.
2. **Contains** – This relationship contains only two foreign keys that is Contest ID and Problem Code which tells us what problem came in which contest.
3. **Attempts** – This relationship contains information about whether a user has attempted a problem or not. This can be used at the time of viewing the history contest details so that user can know about which problems did he attempted. It contains only two attributes i.e. foreign keys Username referencing Username of **User** table and **ProblemCode** referencing ProblemCode of Problems table.

Few Snapshots of Our Interface

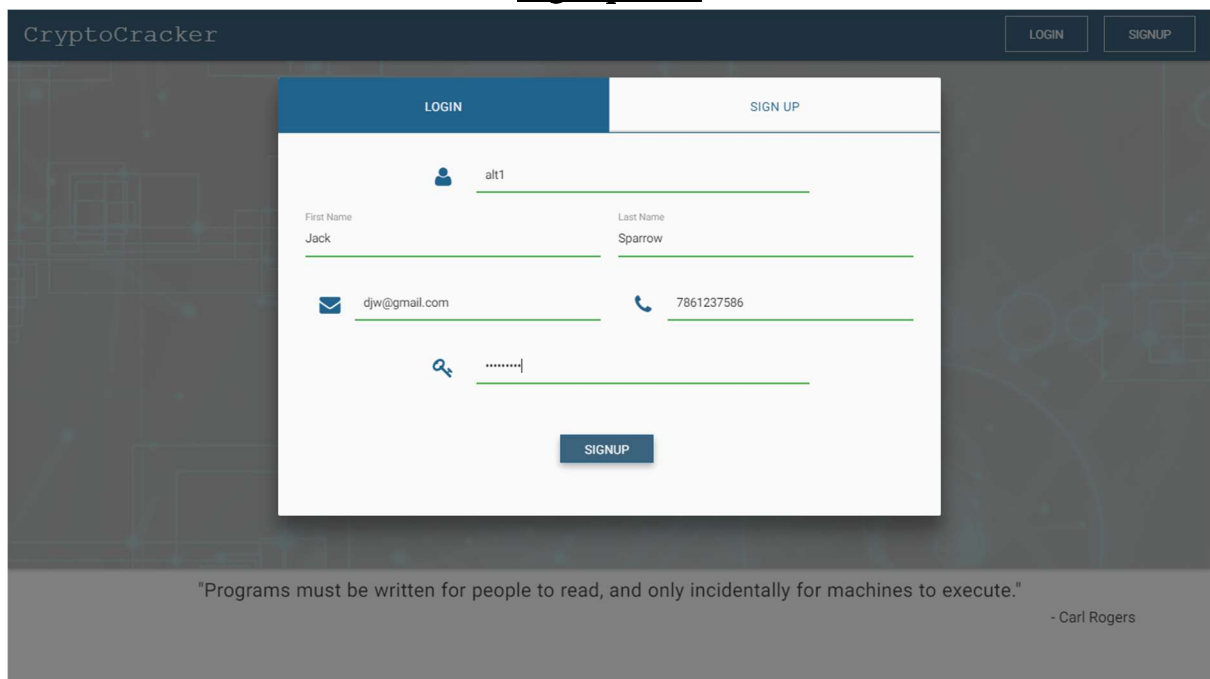
Login Page



"Programs must be written for people to read, and only incidentally for machines to execute."

- Carl Rogers

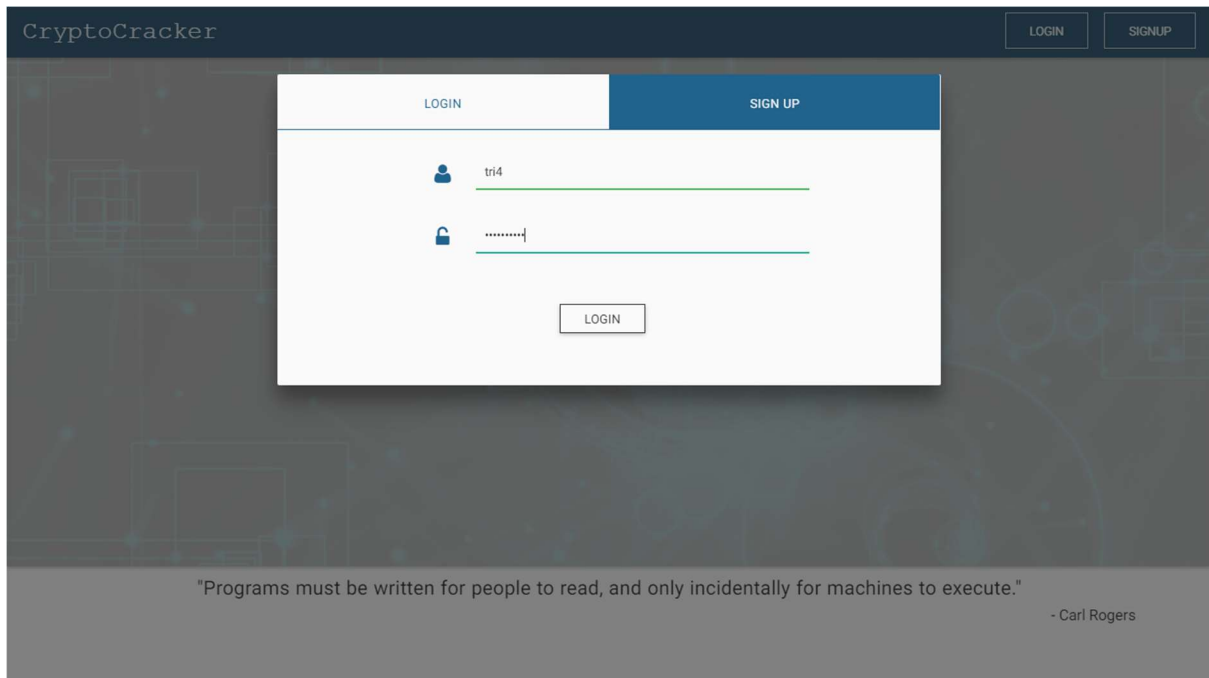
Signup Box



"Programs must be written for people to read, and only incidentally for machines to execute."

- Carl Rogers

Login Box



The screenshot shows the 'CryptoCracker' login interface. At the top, there's a dark blue header with the site name 'CryptoCracker' on the left and 'LOGIN' and 'SIGNUP' buttons on the right. Below the header is a white login box with two tabs: 'LOGIN' (selected) and 'SIGN UP'. The login form contains a username field with the text 'tri4', a password field with masked characters '.....', and a 'LOGIN' button. The background of the page features a dark, abstract pattern of glowing lines and nodes. At the bottom, a quote is displayed: "Programs must be written for people to read, and only incidentally for machines to execute." - Carl Rogers.

CryptoCracker

LOGIN SIGNUP

LOGIN SIGN UP

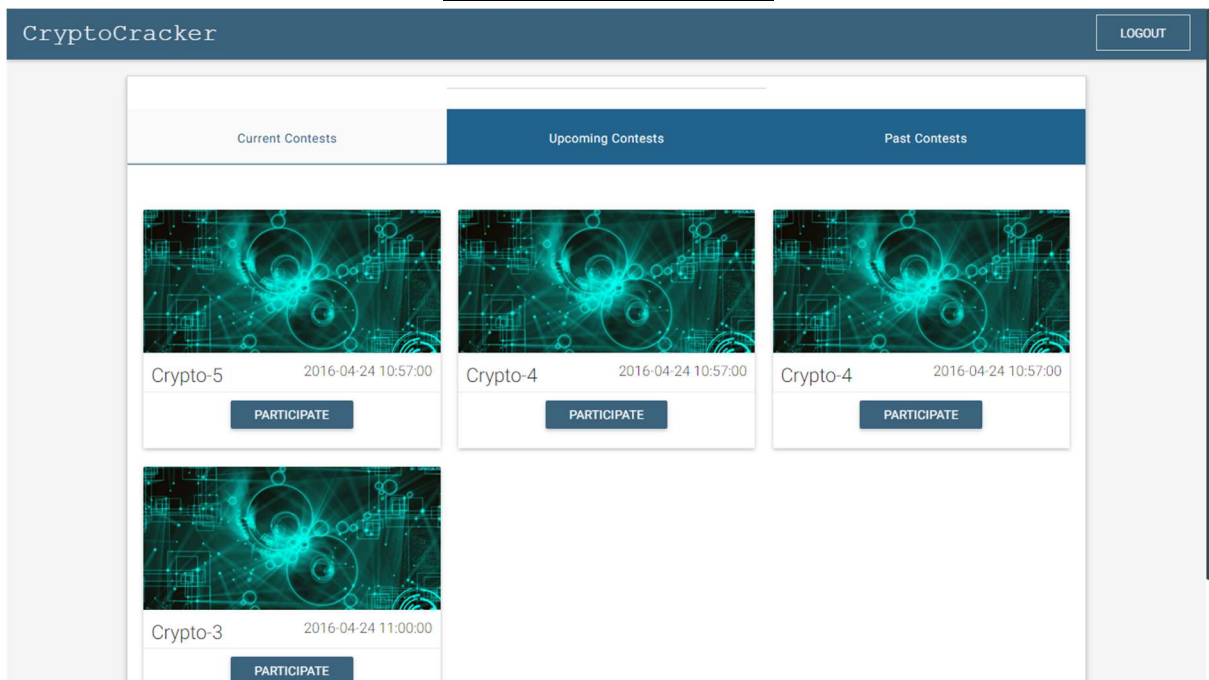
tri4

.....

LOGIN

"Programs must be written for people to read, and only incidentally for machines to execute."
- Carl Rogers

Current Contest Tab



The screenshot shows the 'CryptoCracker' 'Current Contest Tab' interface. At the top, there's a dark blue header with the site name 'CryptoCracker' on the left and a 'LOGOUT' button on the right. Below the header is a white container with three tabs: 'Current Contests' (selected), 'Upcoming Contests', and 'Past Contests'. The 'Current Contests' tab displays a list of contests. Each contest entry includes a green, abstract background image, the contest name (e.g., 'Crypto-5', 'Crypto-4', 'Crypto-3'), the start time (e.g., '2016-04-24 10:57:00', '2016-04-24 11:00:00'), and a 'PARTICIPATE' button.

CryptoCracker

LOGOUT

Current Contests Upcoming Contests Past Contests

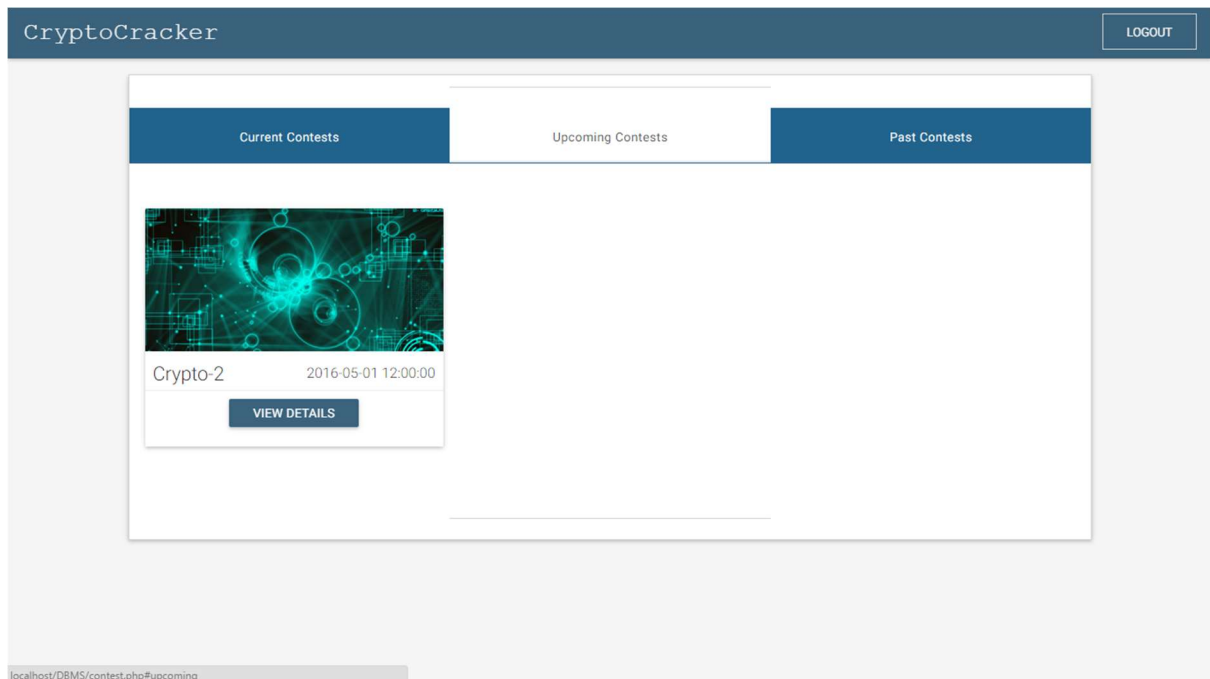
Crypto-5 2016-04-24 10:57:00 PARTICIPATE

Crypto-4 2016-04-24 10:57:00 PARTICIPATE

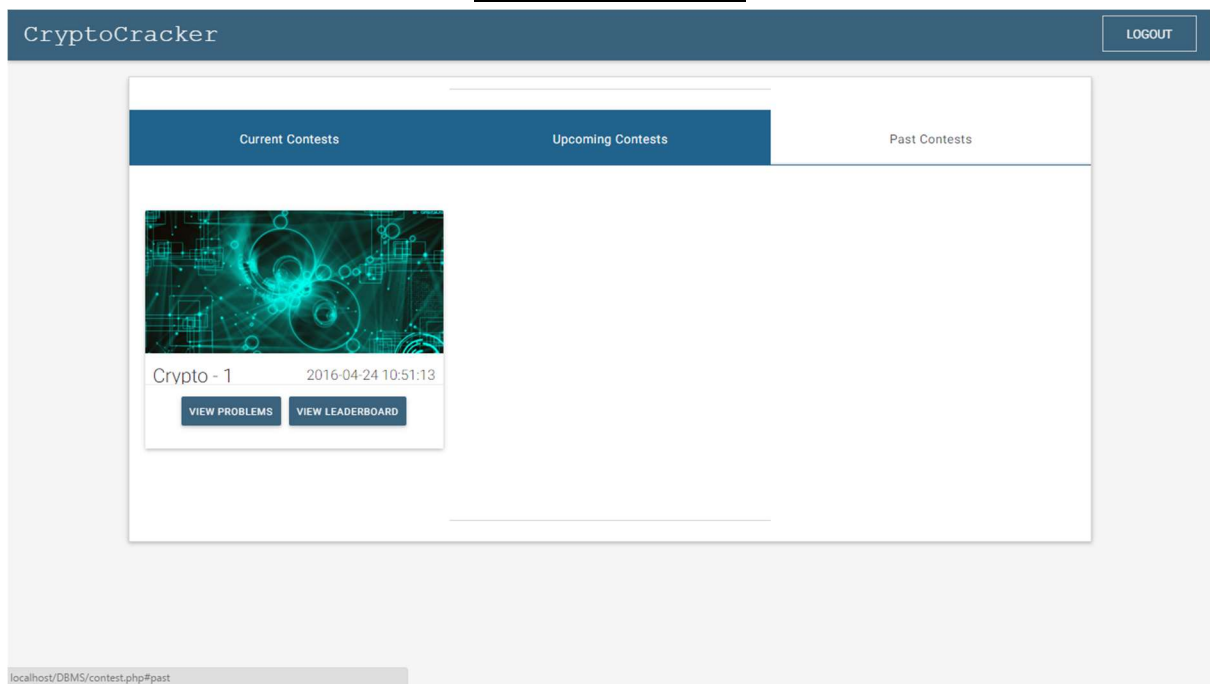
Crypto-4 2016-04-24 10:57:00 PARTICIPATE

Crypto-3 2016-04-24 11:00:00 PARTICIPATE

Upcoming Contest Page



Past Contest Tab



View Details Page

CryptoCracker

LOGOUT

Crypto-5

Description

This is where you code to win
1st Price:1000
2nd Price:750
3rd Price:500

Rules

No rules

Date & Time

2016-04-24 10:57:00

No of Questions

3 Questions

PARTICIPATE

Live Contest Page

CryptoCracker

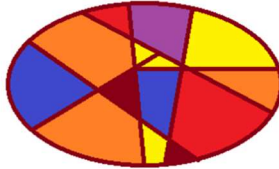
LOGOUT

Questions

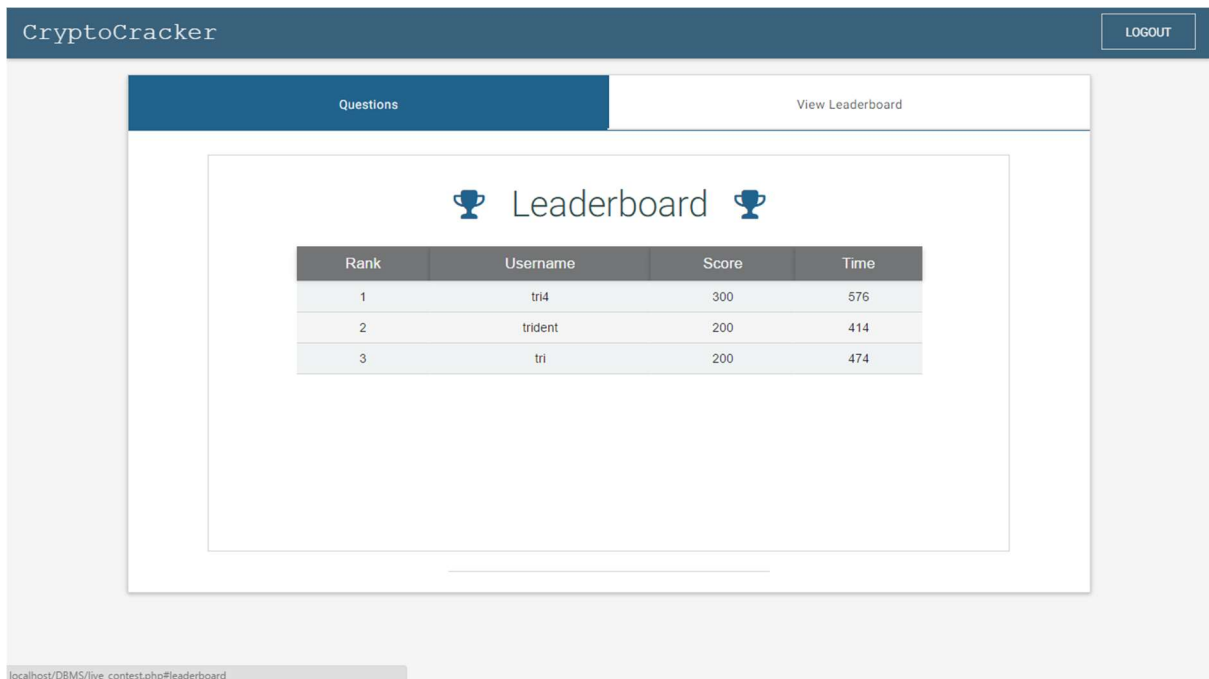
View Leaderboard

1

Karan has his birthday today. He has invited 1000 persons for the party. He is going to cut a cake. Being a hacker there too he thinks of some approach to maximise the number of pieces in minimum cuts. How many minimum cuts do he need to make atleast 1000 pieces of a cake. Assume cake is sufficiently large (circular) and while cutting he crosses the knife completely. (Sizes of cake pieces does not matter, they can be small and large or of any sizes but not definitely equal)



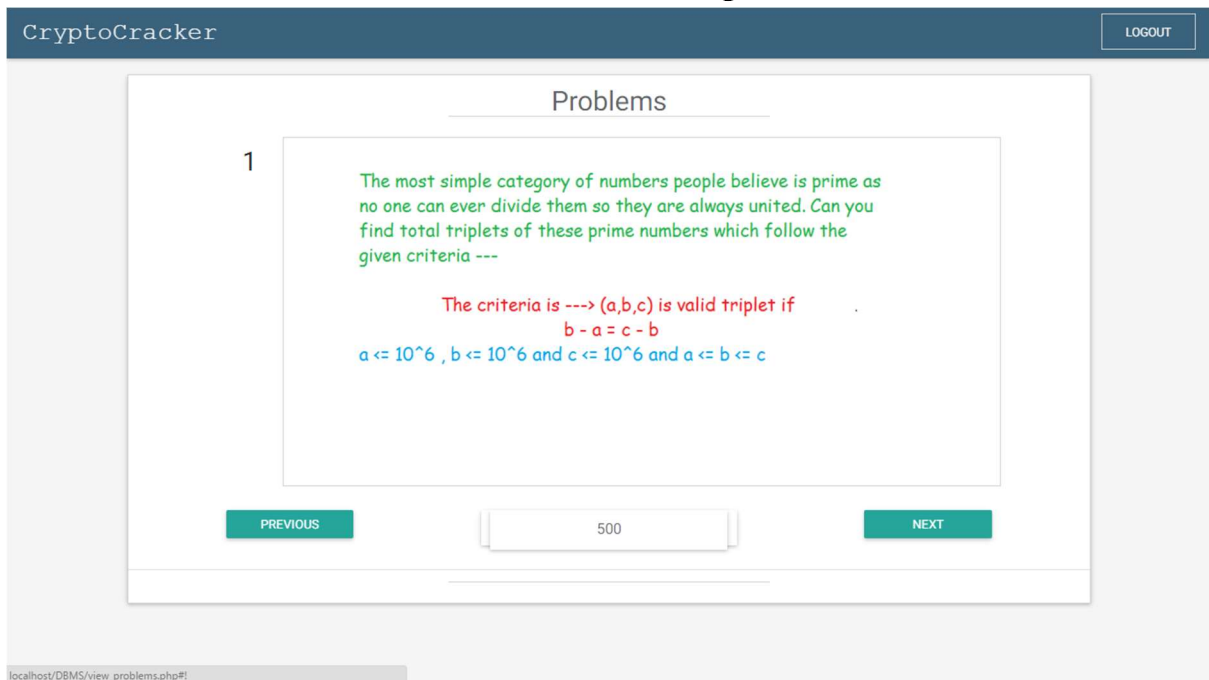
Leader Board Live Tab



The screenshot shows the 'Leaderboard Live Tab' in the CryptoCracker application. The interface has a dark blue header with 'CryptoCracker' on the left and a 'LOGOUT' button on the right. Below the header, there are two tabs: 'Questions' (active) and 'View Leaderboard'. The main content area displays a 'Leaderboard' section with a trophy icon on either side of the title. Below the title is a table with four columns: Rank, Username, Score, and Time. The table lists three users: 'tri4' with Rank 1, Score 300, and Time 576; 'trident' with Rank 2, Score 200, and Time 414; and 'tri' with Rank 3, Score 200, and Time 474. At the bottom left of the page, the URL 'localhost/DBMS/live_contest.php#leaderboard' is visible.

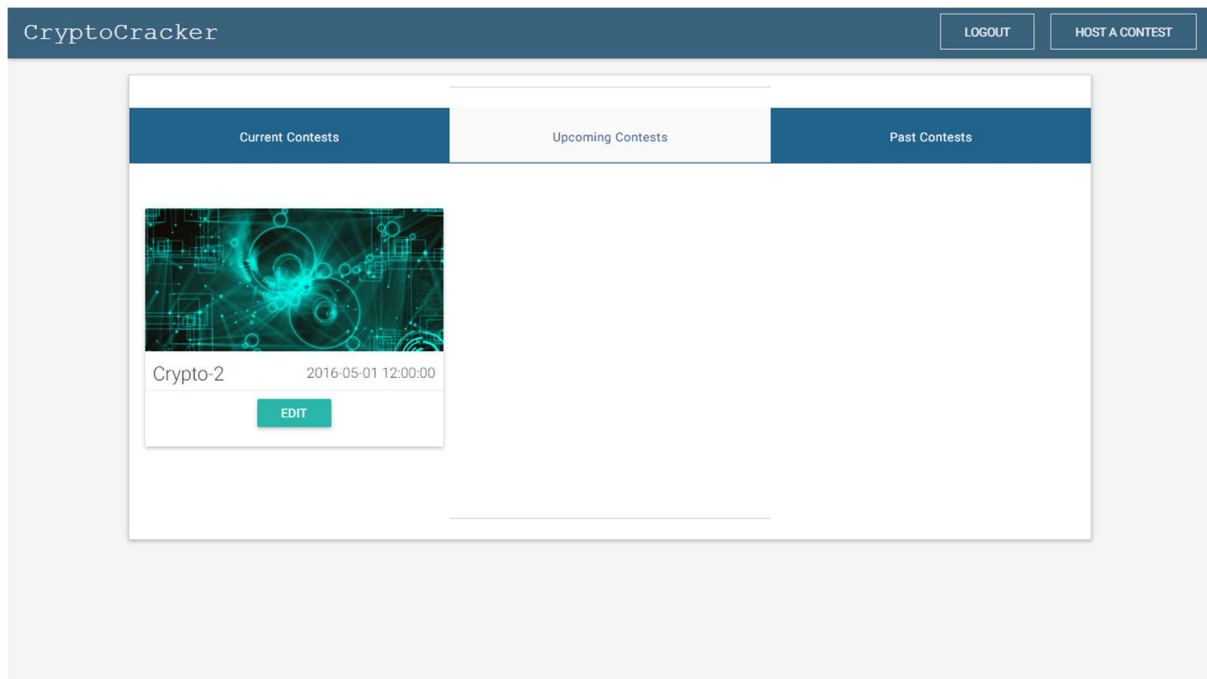
Rank	Username	Score	Time
1	tri4	300	576
2	trident	200	414
3	tri	200	474

Past Contest Page

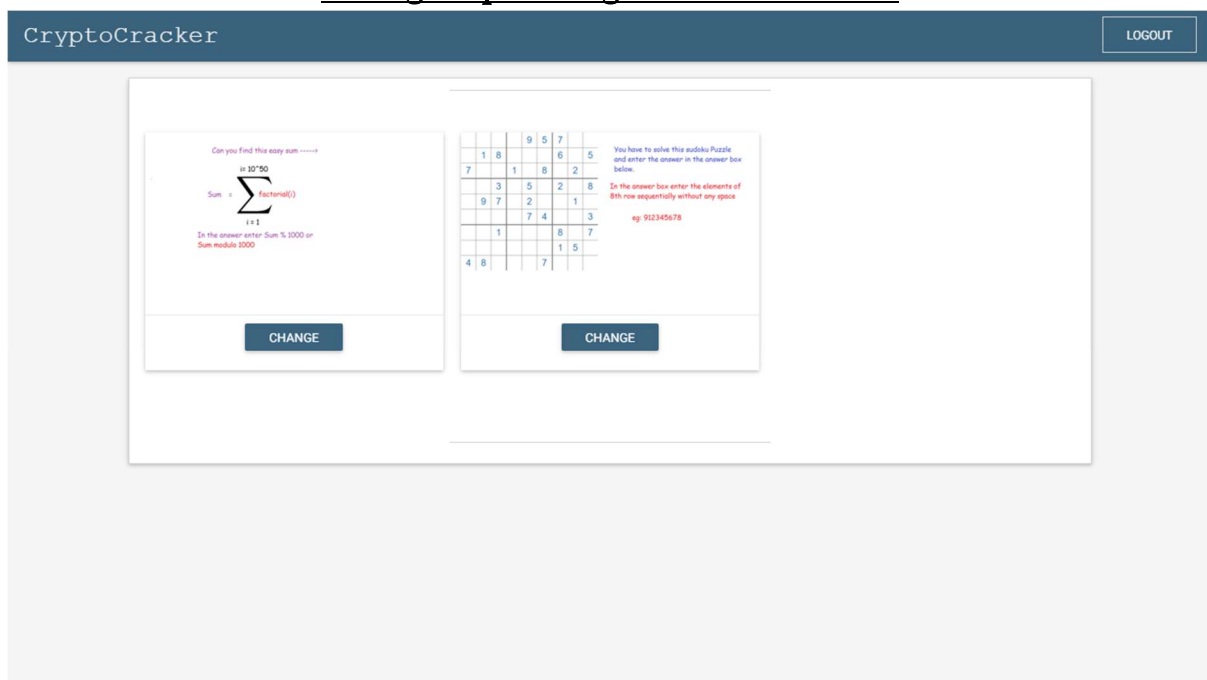


The screenshot shows the 'Past Contest Page' in the CryptoCracker application. The interface has a dark blue header with 'CryptoCracker' on the left and a 'LOGOUT' button on the right. Below the header, there is a 'Problems' section. The first problem, numbered 1, is displayed. The problem text is: 'The most simple category of numbers people believe is prime as no one can ever divide them so they are always united. Can you find total triplets of these prime numbers which follow the given criteria ---'. The criteria are: 'The criteria is ---> (a,b,c) is valid triplet if b - a = c - b'. The constraints are: 'a <= 10^6, b <= 10^6 and c <= 10^6 and a <= b <= c'. Below the problem text, there are three buttons: 'PREVIOUS', '500' (the problem score), and 'NEXT'. At the bottom left of the page, the URL 'localhost/DBMS/view_problems.php#!' is visible.

Upcoming Contest Tab With Edit Access



Change Upcoming Contest Problem



New Uploading of Questions Page

CryptoCracker

LOGOUT

Upload Questions and Answers

1

FILE

13pows.png

Answer 1

1|

SUBMIT

Current Contest Tab

CryptoCracker


LOGOUT

HOST A CONTEST

Current Contests


Upcoming Contests

Past Contests




Crypto-52016-04-24 10:57:00

EDIT




Crypto-42016-04-24 10:57:00

EDIT



Crypto-42016-04-24 10:57:00

EDIT



Crypto-32016-04-24 11:00:00

PARTICIPATE

New Contest Host Page

CryptoCracker

LOGOUT

Fill Contest Details

Contest Name

Debugger

No. of Problems

3

Category

Aptitude

Description

This is demo contest.
No prizes

Rules and Regulations

You have to complete the contest in 3 hours

Start Time

2016-04-24 12:00:00

End Time

2016-04-24 15:00:00

SUBMIT

Upload Questions for New Contest

CryptoCracker

LOGOUT

Upload Questions and Answers

1	<div>FILE</div> a.png	<div>Answer 1</div> 1
2	<div>FILE</div> 13pows.png	<div>Answer 2</div> 2
3	<div>FILE</div> 12erer.png	<div>Answer 3</div> 3

SUBMIT

Confirmation on Contest Setup

CryptoCracker

localhost says:
Your Contest has been Succesfully Set Up

LOGOUT

1

FILE

8.png

1

2

FILE

13pows.png

Answer 2
2

3

FILE

12erer.png

Answer 3
3

SUBMIT

Waiting for localhost...

Database Schema Snapshots

The image displays two screenshots of the phpMyAdmin web interface, showing database schema snapshots for two tables: 'user' and 'registers'.

Top Screenshot: 'user' table

The interface shows the 'user' table selected. The query bar contains the SQL statement: `SELECT * FROM 'user'`. The table structure is displayed below the query bar.

Options	Username	First_Name	Last_Name	Password	Email	Phone	IsAdmin
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	alt1	Jack	Sparrow	backtrack	djw@gmail.com	7861237586	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	altair	Naman	Lal	123456	dancepocket123@gmail.com	9424764887	1
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	alt1r1	Shubhank	Dubey	backtrack1	sd@gmail.com	9384215674	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	apptica	Saurabh	Joshi	123456	technologycorporations@gmail.com	8601251608	1
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	lucky	shubsdja	dubey	shubhash	s@ckj	98023323232	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	tri	samay	Jain	backtrack1	sj@gmail.com	4852941198	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	tri4	Sid	Bac	backtrack1	sb@gmail.com	9576184576	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	trident	Rohit	Rajwani	backtrack1	dp@gmail.com	7837694291	0

Bottom Screenshot: 'registers' table

The interface shows the 'registers' table selected. The query bar contains the SQL statement: `SELECT * FROM 'registers'`. The table structure is displayed below the query bar.

Options	Username	ContestID	Score	Timer
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	tri	1004	200	474
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	tri4	1004	300	576
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	tri4	1005	0	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	trident	1004	200	414

Project Report

The image displays two screenshots of the phpMyAdmin web interface, showing database management for a project named 'CryptoCracker'.

Top Screenshot: Table 'contest'

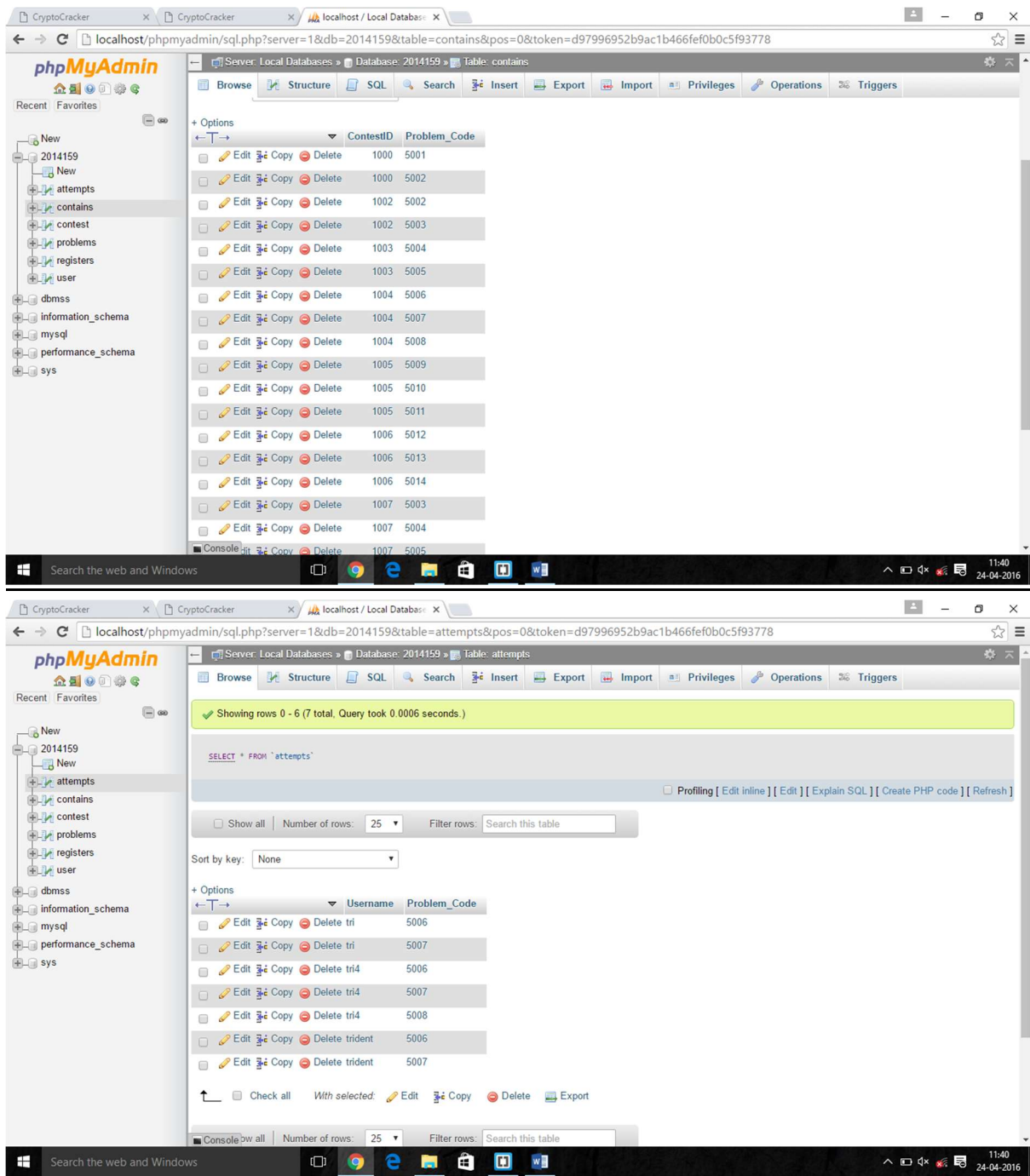
The interface shows the 'contest' table with the following columns: ContestID, ContestName, ContestType, Category, Description, ProblemCount, Admin, StartTime, EndTime, and Rules. The table contains 7 rows of data.

ContestID	ContestName	ContestType	Category	Description	ProblemCount	Admin	StartTime	EndTime	Rules
1000	Crypto - 1	0	1	None	2	applica	2016-04-24 10:51:13	2016-04-24 10:51:13	None
1002	Crypto-2	0	1	This is where you code to win 1st Price: 1000 2nd...	2	altair	2016-05-01 12:00:00	2016-05-10 13:00:00	Just do it.
1003	Crypto-3	0	1	This is where you code to win 1st Price: 1000 2nd...	2	altair	2016-04-24 11:00:00	2016-05-10 13:00:00	Just do no rules
1004	Crypto-5	0	1	This is where you code to win 1st Price: 1000 2nd...	3	applica	2016-04-24 10:57:00	2016-05-10 13:00:00	No rules
1005	Crypto-4	0	1	This is where you code to win 1st Price: 1000 2nd...	3	applica	2016-04-24 10:57:00	2016-05-10 13:00:00	No rules
1006	Crypto-4	0	1	This is where you code to win 1st Price: 1000 2nd...	3	applica	2016-04-24 10:57:00	2016-05-10 13:00:00	No rules
1007	Debugger	0	1	This is demo	3	applica	2016-04-24 12:00:00	2016-04-24 15:00:00	You hav

Bottom Screenshot: Table 'problems'

The interface shows the 'problems' table with the following columns: Problem_Code, Answer, and Max_Score. The table contains 10 rows of data.

Problem_Code	Answer	Max_Score
5001	500	100
5002	500	100
5003	2	100
5004	1	100
5005	2	100
5006	1	100
5007	2	100
5008	3	100
5009	1	100
5010	2	100



Future Aspects

Since our portal will be used to organize a contest then it should be very secure from hackers and other injection attacks. Right now we have tried to block access to all the pages without authentication, but we have not completely blocked the SQL injection attacks. This will be our future goal to improve the project.

To organize events with program submission we have to integrate compiler of common languages in our portal so that apart from quizzes and objective contests we can also host programming challenges for practice and development. Then portal can be used for practicing programming and hosting similar challenges.

We will also provide a profile page so that users can connect to each other and it will become easier to maintain a record of all their achievements in a single option. We hope that this part will be completed as soon as possible.

Conclusion

We created a project that our college clubs can use to organize contests in a very easy and versatile manner. We have created the database by keeping model of the contest in mind. During the project making we learnt a lot of database management and system concepts along with various front end and back end technologies.

In conclusion, a database is a far more efficient mechanism to store and organize data than printouts or paper work. It allows a centralized facility that can easily be modified and quickly shared among multiple users. Having a web based front end removes the requirement of users having to understand and use a database directly, and allows users to connect from anywhere with an internet connection and a basic web browser

References

1. www.hackerearth.com , www.hackerrank.com for understanding features involved in a contest
2. www.w3schools.com for learning various web technologies
3. www.materialize.css for using the css library
4. Database System Concepts – A book by Korth for learning basic concepts involved in a database and formation of ER Model