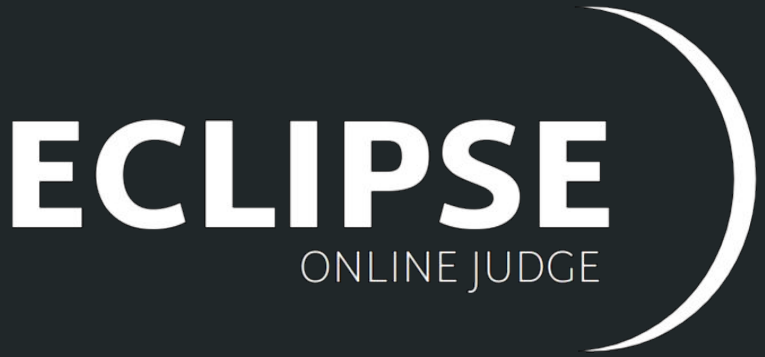

TEAM ECLIPSE

Vighnesh Reddy Konda | Naman Jain | Satti Vamsi Krishna Reddy

ABSTRACT
& PROPOSED IDEA

ABSTRACT

- Create a website for hosting programming contests and maintaining a database of problems in a categorized fashion.
- Login for existing users / Registration facility for new users and storing their details in a database.
- Implement a judge/checker that supports C, C++, Python, Java.
- Implement a Discussion Forum where users can discuss and post regarding various questions and their solutions.
- Implement a proper admin portal, where questions can be uploaded and contests hosted.

ABSTRACT

- Implement a rating feature, rating will get updated after every contest, a statistical approach to calculate ratings appropriately.
- Extend the rating feature to leaderboards ranked according to ratings and categorised in various ways (according to say institution).
- Prevent execution of potential ‘malware/untrusted’ code that can be uploaded by a user intentionally.

PROPOSED IDEA

We had proposed to implement the following features in Eclipse OJ:

1. Create a website for hosting programming contests and maintaining a database of problems in a categorized fashion.

2. Login for existing users / Registration facility for new users and storing their details in a database.

3. Implement a judge/checker that supports C, C++, Python, Java.

4. Implement a Discussion Forum where users can discuss and post regarding various questions and their solutions.

5. Implement a proper admin portal, where questions can be uploaded and contests hosted.

6. Implement a rating feature, rating will get updated after every contest, a statistical approach to calculate ratings appropriately

7. Extend the rating feature to leaderboards ranked according to ratings and categorised in various ways (according to say institution).

8. Prevent execution of potential ‘malware/untrusted’ code that can be uploaded by a user intentionally.

9. Implement a private messaging interface which also support file-transfers for users to interact and collaborate.*

*To be done if time permits

INTRODUCTION

What’s an OJ (Online Judge)?

- An online judge is an online system to test programs in programming contests. They are also used to practice for such contests.
- The system can compile and execute code, and test them with pre-constructed expected output data.
- Submitted code may be run with restrictions, including time limit, memory limit, security restriction and so on.
- The output of the code will be captured by the system, and compared with the standard output. The system will then return the result.
- Online Judges have rank lists based on user ratings

Source: Wikipedia

MOTIVATION

Why Eclipse OJ?

- Most of the popular online judges today ask for a price to hold a contest.
- No big and famous ‘open-source OJs’ - so felt the need of one :P
- Additionally IIT Bombay doesn’t have an online judge of it’s own, so we decided to make a state-of-the-art Online judge for our institute, which we strongly believe will increase the community of sport programmers in our insti.
- Team collaboration has been difficult, specifically for learners initially. Messaging systems (if any) already seem to be naive and not user-friendly.

MAJOR SOFTWARES & TOOLS USED

1. Project Development
2. Project Management
3. Django Packages

SOFTWARE REQUIREMENTS

FOR PROJECT DEVELOPMENT

- Django 1.11.5
- HTML5, CSS3 (using Sass)
- Javascript (including jQuery, AJAX)
- Docker
- BASH
- Others like: MathJax, Bootstrap4 ... *etc.*

FOR PROJECT MANAGEMENT

- Git (for Version Control)
- LaTeX (Documentation)

DJANGO PACKAGES

- django-bootstrap4
- django-macros
- django-countries

(OPEN-SOURCE) REQUIREMENTS & SETUP

INSTALLING SOFTWARES

- Python is a must for installing Django and we assume that both the latest versions Django 1.11.5 and Python 3.6.3 (at the time of this writing) are installed in your PC.
- To install django-bootstrap4, run: pip3 install django-bootstrap4
- To install django-countries, run: pip3 install django-countries
- To install django-macros, run: pip3 install django-macros
- Install Sass and Docker CE from the respective websites.

SETUP DOCKER

- We need to create a container named ’dock_container’ which our OJ uses to sandbox the code before executing them.
- Step 1: Move to the Project folder using cd
- Step 2: Run docker build -t ubuntu:v1 'path to project folder'/Dockerfile
- Step 3: Run docker create -id ubuntu:v1 --name dock_container

Note: You may need to start the dock_container everytime you restart the PC. You may automate this process easily however.

Yayy! And thats it, we are ready to run our project!

OUR IMPLEMEN-TATIONS

1. Django
2. Sandboxing
3. Ranking System
4. Ajax
5. Ace Editor

Django

All the frontend UI for our website is done using HTML, CSS, JS , Bootstrap.

• All the various modules like login, registration, discussion forum, database of questions, leaderboard, messaging ...etc are done as ‘separate apps’ thus accounting for reusability.

• We would implement a basic login/registration page with Django, using Django’s authentication packages at ‘django.contrib.auth’

• We plan to use Django’s argon2 hashing algorithm to provide secure hashing for passwords present at (django.contrib.auth.hashers)

- Customize the ‘admin.py’ in Django specifically for adding problems and hosting contests in a simple way using sqlite3 databases.

- All the submission requests will be handled via. a queue, hence preventing any possible server crash.

- Every code is sandboxed using Docker to avoid any harm done by running untrusted/malicious codes.

- Implement a Rating system that reflects the user’s performance based on a simplified Elo Rating System.

Sandboxing

We intend to prevent any harm to server caused by execution of ‘untrusted code’. This is commonly called ‘Sandboxing’ in Computer Security. At first we chose to implement a Chroot Jail but we found using Docker was much simpler and elegant So, we decided to use Docker

All the user submitted codes are executed in a container called dock_container. We copy the testcase folder into the container and do the testing. So if a user sends any malicious code if at all something happens the container gets effected rather than our main sy

Ranking System

Our rating system is similar to Elo rating system. the calculations part is simplified as below:

Performance-Rating = (total-of-opponent-ratings + 400 × (Wins − Losses))/Games

• The Elo rating system is a method for calculating the relative skill levels of players in multiplayer type of games.

• We found that Elo rating system is most appropriate for us and also Codechef and Hackerrank implement their rating system mainly based on Elo (though not entirely the same)

Ajax

We have used Ajax JsonResponse method to create an Ascynchronous HTTP request in the signup form. Once the username input box looses focus, ajax sends an ajax request to server containing the preferred username by the user. This is checked with the existing user database and if there exists such an user, the user is notified dynamically

Ace Editor

We linked the editor which the user can directly use to code. He can check with his own custom testcases. He can directly submit his code to a problem. The text editor supports C,C++,Python2,Python3 and Java with syntax highlighting. The editor can be customized by changing its theme. We provide one Dark and one Light Theme

DESIGN AESTHETICS

ECLIPSE

ONLINE JUDGE

Log in

Sign Up

Sign Up

Username

Username

Your username should contain only letters, numbers and symbols including _ @ . + -

Password

Password

Password confirmation

Password confirmation

First name

First name

Last name

Last name

Email

Country

Afghanistan

City

City

Institute

Institute

Sign Up

Instructions

Your password must contain at least 8 characters, can't be entirely numeric and should contain only letters, numbers and symbols including _ @ . + -

PROFILE

Friends | Coders | Learners

John Doe

(@normal1)

Change Password Edit Profile

Rating	1690
Country	Bahamas
City	Laos
Institute	Rice University
Email	normal1@gmail.com

CONTESTS

Code | Compete | Excel

List of Contests

#	STATUS	NAME	START	END
1	Completed	The First	Oct. 26, 2017, 5:11 p.m.	Oct. 26, 2017, 5:15 p.m.
2	Completed	Second Contest	Oct. 26, 2017, 6 a.m.	Oct. 26, 2017, 6 p.m.

Top Rated

#	User	Rating
1	tourist	3100
2	V_oo_V	3025
3	-XraY-	2986

LEADERBOARD

Get | Set | Go

Filter

Username

Username

Country

Country

City

City

Institute

Institute

Search

Leaderboard

RANK	USERNAME	COUNTRY	CITY	INSTITUTE	RATING
1	normal1	Bahamas	Laos	Rice University	3120
2	normal2	Ecuador	London	IITK	3190
3	normal3	India	Mumbai	IITBombay	2900

DISCUSS

Ask | Answer | Learn

Add new post

Editorial for First Contest

Posted by Shah Kumar on Oct. 26, 2017, 5:09 p.m.

lorem ipsum

Second Post

Posted by Shah Kumar on Oct. 26, 2017, 5:10 p.m.

lorem lo...

IIT Bombay

Posted by First User Khan on Oct. 29, 2017, 10:53 a.m.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. I...

SUBMISSIONS

Code | Submit | Repeat

Submission List

#	Submission Time	User	Problem	Language	Verdict
5	Oct. 29, 2017, 11:34 a.m.	normal1	The First Problem	C++	Accepted
4	Oct. 29, 2017, 11:34 a.m.	normal2	The First Problem	Python3	Compilation Error
3	Oct. 29, 2017, 11:33 a.m.	normal3	The First Problem	Java	Time Limit Exceeded
2	Oct. 29, 2017, 11:32 a.m.	normal1	The First Problem	C++	Runtime Error
1	Oct. 26, 2017, 5:14 p.m.	normal3	The First Problem	C	Wrong Answer

FINAL DELIVERABLES

Final deliverables

- Create a website for hosting programming contests and maintaining a database of problems in a categorized fashion.
- Login for existing users / Registration facility for new users and storing their details in a database.
- Implement a judge/checker that supports C, C++, Python2, Python3, Java.
- Implement a Discussion Forum where users can discuss and post regarding various questions and their solutions.
- Implement a proper admin portal, where questions can be uploaded and contests hosted.

- Implement a rating feature, rating will get updated after every contest, a statistical approach to calculate ratings appropriately.
- Extend the rating feature to leaderboards ranked according to ratings and categorised in various ways (according to say institution).
- Prevent execution of potential ‘malware/untrusted’ code that can be uploaded by a user intentionally using Docker.
- Linked Ajax Ace editor where user can directly code and check with his custom inputs or submit for a problem

POSSIBLE EXTENSIONS

Possible Extensions

- Messaging System with support for group chats
- Automatic Test Case generation for problems depending on the constraints
- Virtual contests where a user can take a past contest like a current contest though his rating won’t get updated
- Better rating system that takes into the consideration of the problem weights

CONCLUSION

Conclusion

We learned the main features of Django, Sandboxing using Docker, database relationships using sqlite3, and improved our knowledge of collaboration with Git and how open source projects are really developped, Bash

THANK YOU!