

A  
Mini Project  
On  
**CODING CONTESTS DETECTION USING CLIST API**

(Submitted in partial fulfillment of the requirements for the award of Degree)

**BACHELOR OF TECHNOLOGY**

In  
**COMPUTER SCIENCE AND ENGINEERING**

By  
**ANKIT PATEL (197R1A05J5)**

Under the Guidance of

**A.KIRAN KUMAR**

(Assistant Professor)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CMR TECHNICAL CAMPUS**

**UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New  
Delhi) Recognized Under Section 2(f) & 12(B) of the UGC Act. 1956, Kandlakoya (V),

Medchal Road, Hyderabad-501401.

**2019-2023**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**CERTIFICATE**

This is to certify that the project entitled “**CODING CONTESTS DETECTION USING CLIST API**” being submitted by **ANKIT PATEL(197R1A05J5)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the CMR Technical Campus, is a record of bonafide work carried out by them under our guidance and supervision during the year 2022-2023.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**A KIRAN KUMAR**  
(Assistant Professor)  
INTERNAL GUIDE

**Dr. A. Raji Reddy**  
DIRECTOR

**Dr. K. Srujan Raju**  
HOD

**EXTERNAL EXAMINER**

**Submitted for viva voice Examination held on \_\_\_\_\_**

## ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **A.Kiran Kumar**, Assistant Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr. Punyaban Patel, Ms. Shilpa, Dr. M . Subha Mastan Rao & J. Narasimharao** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju**, Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy**, Director for being cooperative throughout the course of this project. We also express our sincere gratitude to **Sri. Ch. Gopal Reddy**, Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

**ANKIT PATEL (197R1A05J5)**

## **ABSTRACT**

We always miss coding contests now and then, it feels bad to miss a contest when your placement season is coming and you want to be prepared. We always wonder if there could be any website/app that shows all platform contests in one place so we could never miss a coding contest.

There are many coding platforms nowadays and as we want to practice more and more we join in all the contests running on the various platforms. Coding contest on various platform is at different time and for that, we need to check all coding platform, again and again, to know when the contest is there this waste a lot of time. Due to this, we miss a lot of contests as we are not aware of when and on which platform contest is going on. This comes as a hurdle when we are preparing and practicing for our college placement.

So, I proposed a website coding contests detection using cList api Which tracks all the coding platform contests. So, never miss another coding contest from now. This will track all the past contests, present contests, and ongoing contests of all the coding platforms. This helps to see all the coding platforms such as Hackerrank, HackerEarth, Codeforce, Codechef, Leetcode, At coder, and many more coding platforms.

## LIST OF FIGURES/TABLES

FIGURE NO	FIGURE NAME	PAGE NO
Figure 3.1	Project architecture for coding contests detection using clist api	7
Figure 3.2	Usecase diagram for coding contests detection using clist api	8
Figure 3.3	Class diagram for coding contests detection using clist api	9
Figure 3.4	Sequence diagram for coding contests detection using clist api	10
Figure 3.5	Activity diagram for coding contests detection using clist api	11

## **LIST OF SCREENSHOTS**

<b>SCREENSHOT NO.</b>	<b>SCREENSHOT NAME</b>	<b>PAGE NO.</b>
Screenshot 5.1	Coding Platforms	16
Screenshot 5.2	Route(live/past/future contest)	16
Screenshot 5.3	Individual contests card	17
Screenshot 5.4	Add to calendar	17
Screenshot 5.5	Visit the Calendar	18

# TABLE OF CONTENTS

<b>ABSTRACT</b>	i
<b>LIST OF FIGURES</b>	ii
<b>LIST OF SCREENSHOTS</b>	iii
<b>1. INTRODUCTION</b>	1
1.1 PROJECT SCOPE	1
1.2 PROJECT PURPOSE	1
1.3 PROJECT FEATURES	1
<b>2. SYSTEM ANALYSIS</b>	2
2.1 PROBLEM DEFINITION	2
2.2 EXISTING SYSTEM	2
2.2.1 LIMITATIONS OF THE EXISTING SYSTEM	3
2.3 PROPOSED SYSTEM	3
2.3.1 ADVANTAGES OF PROPOSED SYSTEM	3
2.4 FEASIBILITY STUDY	4
2.4.1 ECONOMIC FEASIBILITY	4
2.4.2 TECHNICAL FEASIBILITY	5
2.4.3 SOCIAL FEASIBILITY	5
2.5 HARDWARE & SOFTWARE REQUIREMENTS	5
2.5.1 HARDWARE REQUIREMENTS	5
2.5.2 SOFTWARE REQUIREMENTS	6
<b>3. ARCHITECTURE</b>	7
3.1 PROJECT ARCHITECTURE	7
3.2 DESCRIPTION	7
3.3 USE CASE DIAGRAM	8
3.4 CLASS DIAGRAM	9
3.5 SEQUENCE DIAGRAM	10
3.6 ACTIVITY DIAGRAM	11
<b>4. IMPLEMENTATION</b>	12
4.1 SAMPLE CODE	12-15
<b>5. SCREENSHOTS</b>	16-18

<b>6. TESTING</b>	19
6.1 INTRODUCTION TO TESTING	19
6.2 TYPES OF TESTING	19
6.2.1 UNIT TESTING	19
6.2.2 INTEGRATION TESTING	20
6.2.3 FUNCTIONAL TESTING	20
6.3 TEST CASES	21
6.3.1 CLASSIFICATION	21
<b>7. CONCLUSION &amp; FUTURE SCOPE</b>	22
7.1 PROJECT CONCLUSION	22
7.2 FUTURE SCOPE	22
<b>8. REFERENCES</b>	23
8.1 REFERENCES	23
8.2 GITHUB LINK	23



# **1. INTRODUCTION**

# **1. INTRODUCTION**

## **1.1 PROJECT SCOPE**

This project is titled “Coding Contests detection using clist api”. We always miss coding content now and then, it feels bad to miss a contest when your placement season is coming and you want to be prepared. We always wonder if there Could be any website/app that shows all platform contests in one place so we would never miss a coding contest again.

## **1.2 PROJECT PURPOSE**

This website coding contests detection using cList api Which tracks all the coding platform contests. So, never miss another coding contest from now. This will track all the past contests, present contests, and ongoing contests of all the coding platforms. This helps to see all the coding platforms such as Hackerrank, HackerEarth, Codeforce, Codechef, Leetcode, At coder, and many more coding platforms in one place.

## **1.3 PROJECT FEATURES**

The main features of this project are we can keep track of all the coding contests going on various coding platforms. The 3 types of contests are split across 3 different routes. Each of them is different color coded for better user understanding. Each contest has a link with which you can directly visit the specific programming site. This gives an extra advantage of finding all the contests which are running on different coding platforms. Thus it reduces a lot of manual effort in searching for contests on these websites. This helps a lot of students while they are preparing for their placement and they will not miss any coding contests.

## **2. SYSTEM ANALYSIS**

## **2. SYSTEM ANALYSIS**

### **SYSTEM ANALYSIS**

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

#### **2.1 PROBLEM DEFINITION**

A general statement of coding contests detection using clist api is based on a real-life problem of a student who is preparing for placaments. During placement, students forget to give contests, which leads to effects in their placement preparation. To avoid this, I proposed this project.

#### **2.2 EXISTING SYSTEM**

There are many coding platforms nowadays and as we want to practice more and more we join in all the contests running on the various platforms. Coding contest on various platform is at different time and for that, we need to check all coding platform, again and again, to know when the contest is there this waste a lot of time. Due to this, we miss a lot of contests as we are not aware of when and on which platform contest is going on. This comes as a hurdle when we are preparing and practicing for our college placement.

### **2.2.1 DISADVANTAGES OF EXISTING SYSTEM**

- There is no such proper platform to keep track of all coding contests.
- We need to visit each site to check the contest timing which wastes our time.
- We always miss a contest because we are not aware of that contest's timing

To avoid all these limitations and make the work more accurate the system needs to be implemented efficiently.

### **2.3 PROPOSED SYSTEM**

I propose a site where we can keep track of all the coding contests going on various coding platforms. The 3 types of contests are split across 3 different routes. Each of them is different color coded for better user understanding. Each contest has a link with which you can directly visit the specific programming site. This gives an extra advantage of finding all the contests which are running on different coding platforms. Thus it reduces a lot of manual effort in searching for contests on these websites. This helps a lot of students while they are preparing for their placement and they will not miss any coding contests.

#### **2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM**

- All coding platform contests details in a single place.
- No time-consuming process of visiting all coding platforms again and again.
- Get all the updates of upcoming, live, and past coding contests.
- Add the upcoming contest to your calendar and get a notification when the contest is live.

## 2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis:

- EconomicFeasibility
- TechnicalFeasibility
- SocialFeasibility

### 2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminaryinvestigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication that the system is economically possible for development.

## **2.4.2 TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

## **2.4.3 BEHAVIORAL FEASIBILITY**

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible

## **2.5 HARDWARE & SOFTWARE REQUIREMENTS**

### **2.5.1 HARDWARE REQUIREMENTS:**

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

- Processor : Intel Dual Core I5
- Hard disk : 40GB
- RAM : 4GB
- Input devices : Keyboard, mouse.

## 2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

- Operating system : Windows 10
- Languages : Html, CSS, Html, CSS, JS, ReactJS
- Tools : VS Code Editor, Chrome Browser



### **3. ARCHITECTURE**

### 3. ARCHITECTURE

#### 3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final prediction.

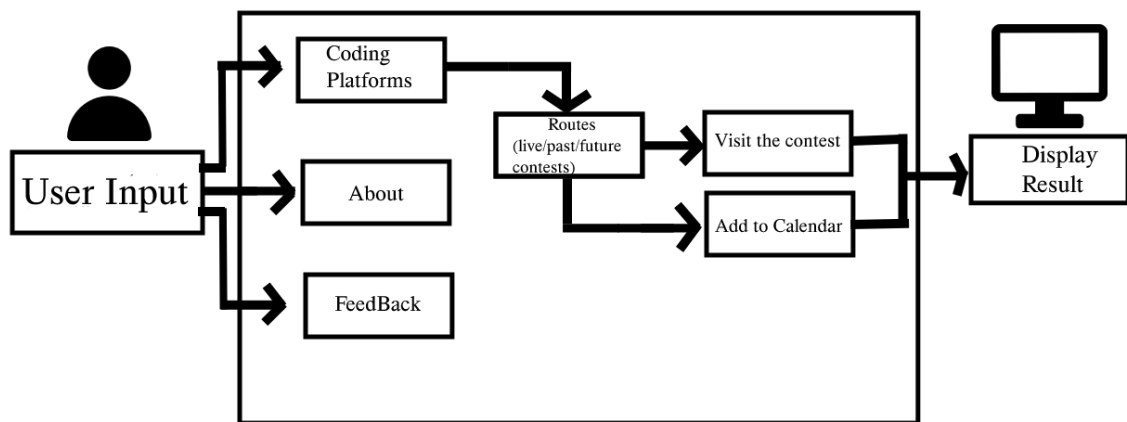


Figure 3.1: Architecture for Coding Contests detection using clist api

#### 3.2 DESCRIPTION

So, I proposed a website coding contests detection using cList api Which tracks all the coding platform contests. So, never miss another coding contest from now. This will track all the past contests, present contests, and ongoing contests of all the coding platforms. This helps to see all the coding platforms such as Hackerrank, HackerEarth, Codeforce, Codechef, Leetcode, At coder, and many more coding platforms in one place.

### 3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model.

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

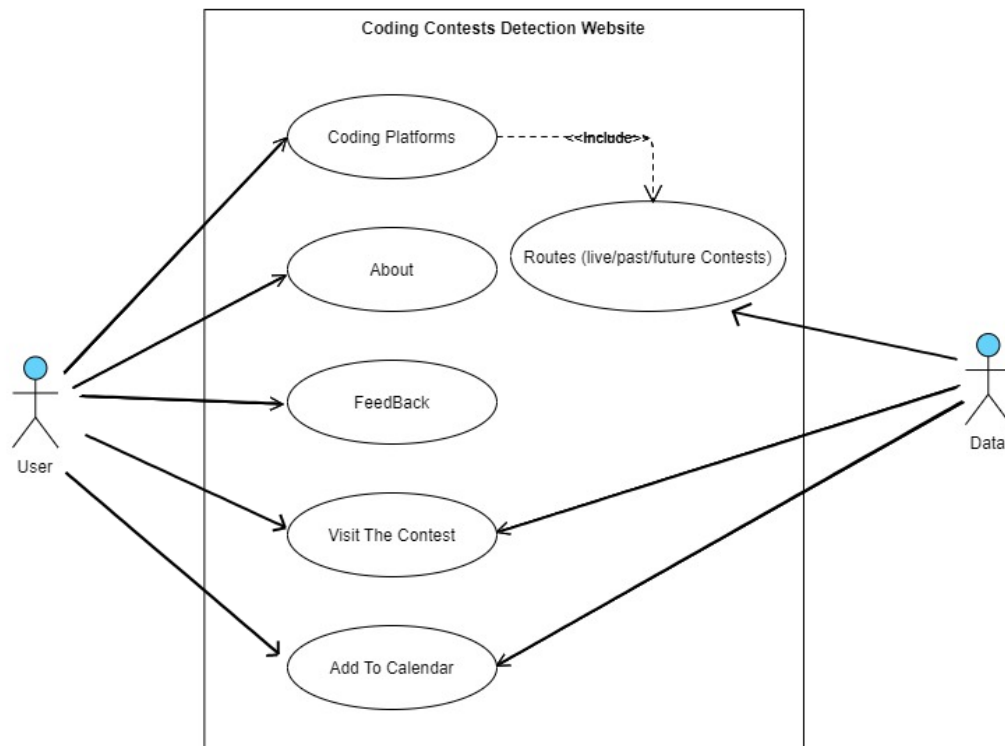


Figure 3.2: Use Case Diagram for coding contests detection using clist api

### 3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations(or methods), and the relationships among objects.

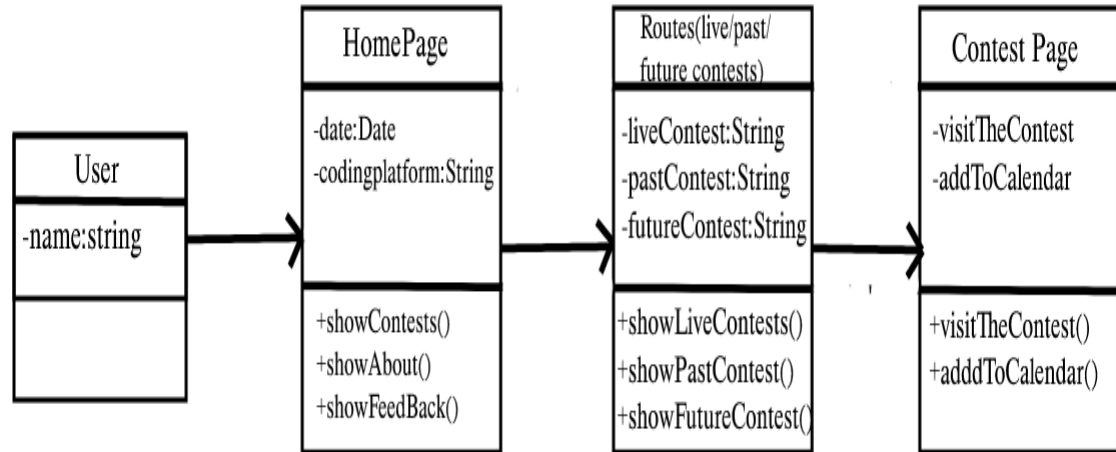


Figure 3.3: Class Diagram for coding contests detection using clist api

### 3.5 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

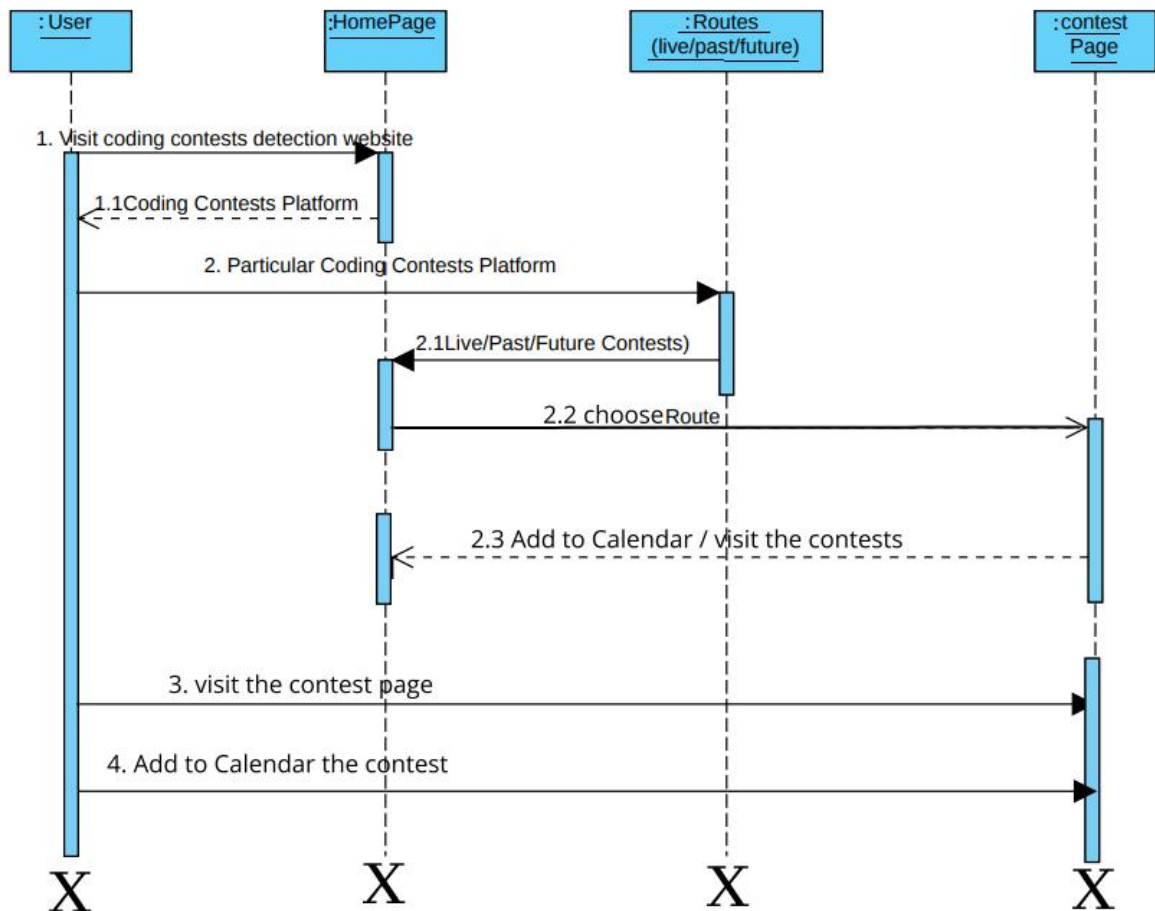


Figure 3.4: Sequence Diagram for coding contests detection using clist api

### 3.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more data stores.

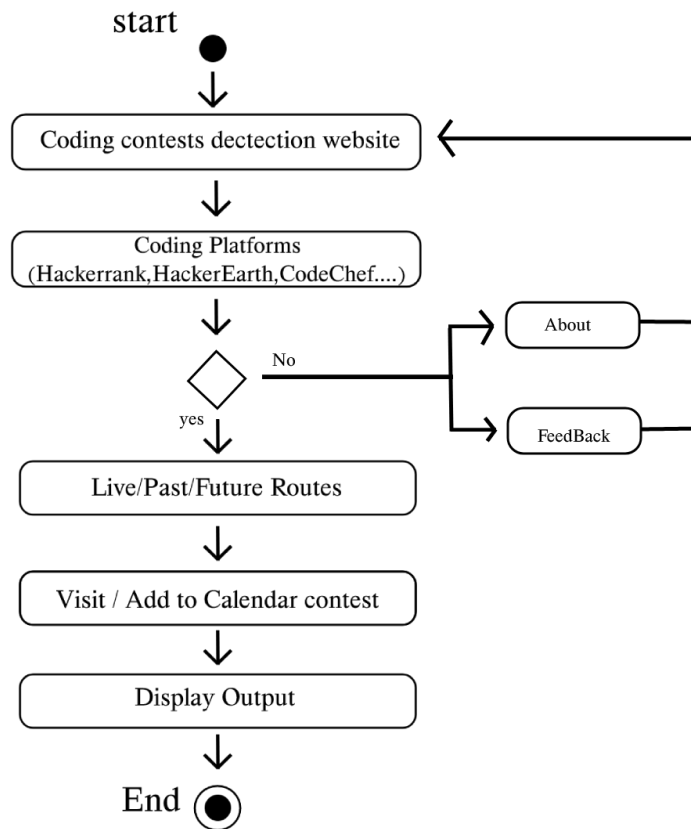


Figure 3.5: Activity Diagram for coding contests detection using clist api

## **4. IMPLEMENTATION**

## 4.IMPLEMENTATION

### 4.1 SAMPLE CODE

#### App.js

```
import './App.css';
import Home from './components/Home';
import Navbar from './components/Navbar';
import Contest from './components/Contest'

// import useFetch from "react-fetch-hook"
import { BrowserRouter as Router ,Routes,Route} from "react-router-dom"
function App() {
  return (
    <Router>
      <Navbar/>
      <Routes>
        <Route>
          <Route path="/" element={ <Home/> }></Route>
          <Route path="/about" element={ <About/> }></Route>
          <Route path="/feedback" element={ <Contact/> }></Route>
          <Route path="/:contestname" element={ <Contest/> }></Route>
        </Route>
      </Routes>
    </Router>
  );
}

export default App;
```



**Data.js**

```

export const platformList = [
  {
    id: "1",
    name: "codeforces",
    link:"codeforces",
    url:"https://res.cloudinary.com/practicaldev/image/fetch/s--. jpg",
  },
  {
    id: "2",
    name: "codechef",
    link:"code_chef",
    url:"https://pbs.twimg.com/profile_images/1477930785537605633/ROTVNVz7_400x400.jpg"
  },
  {
    id: "3",
    name: "Leetcode",
    link:"leet_code",
    url:"https://leetcode.com/static/images/LeetCode_Sharing.png"
  },
  {
    id: "4",
    name: "HackerRank",
    link:"hacker_rank",
    url:"https://upload.wikimedia.org/wikipedia/commons/4/40/HackerRank_Icon-1000px.png"
  },
];

```

**Contests.js**

```

import React,{useState,useEffect} from 'react'
import {useParams} from "react-router";
import Card from './Card';
function Contest() {
  const {contestname} = useParams();
  const [contestData, setContestData] = useState([]);
  const [cData, setCData] = useState([]);
  return (
    <div className='contestnames'>
      <div className='nav2'>{contestname}</div>
      <div className='head'>
        <a href='#in-24-hours' onClick={()=>handlepastClick("Yes")}>
          in 24 hours
        </a>
        <a href='#live' onClick={()=>handleliveClick("CODING")}>
          live
        </a>
        <a href='#future' onClick={()=>handlefutureClick("BEFORE")}>
          future
        </a>
      </div>
      {(cData.length==0)?<EmptyList/>:<Card cData={cData} val={val}/>}
    </div>
  )
}

export default Contest

```

**Cards.js**

```

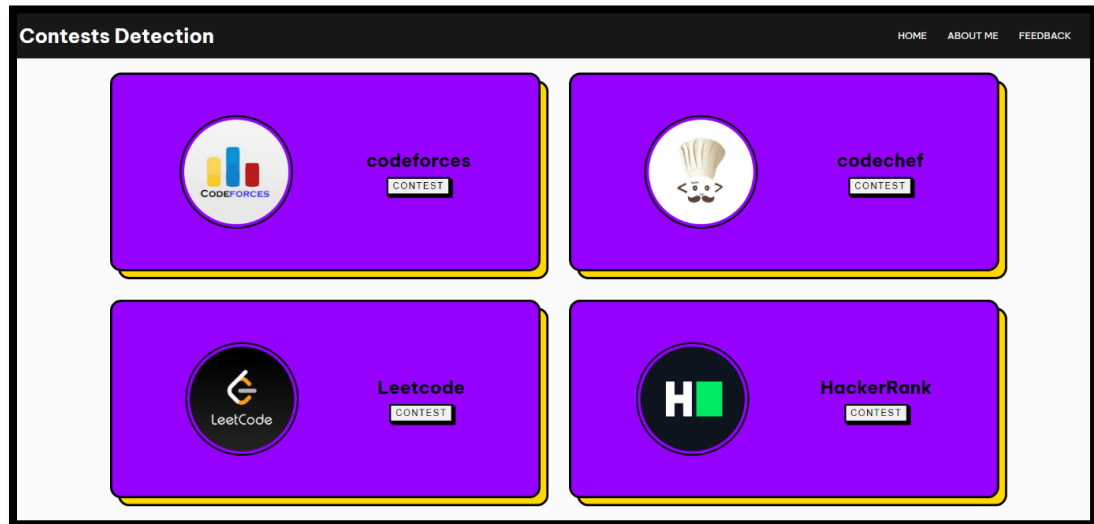
import React, { useEffect } from 'react'
import './card.css'
import { atcb_action, atcb_init } from 'add-to-calendar-button';
function Card({ cData, val }) {
  const changeName = e => {
    setName(e.target.value);
  };
  useEffect(()=>{
    setInterval(showDate,1000);
  },[diftime]);
  const showDate=()=>{
    setDifTime(new Date());
  }
  return (
    <div className='card'>
      {
        cData.map((item)=>
          <div className='card-child'>
            <h1 className='item-name'> <b>{item.name.toUpperCase()}</b></h1>
            <h1 className='duration'> <b>Duration :</b> {item.duration/60}mins</h1>
            <h1 className='start-date'> <b>Start Date: </b>`${new
Date(item.start_time).toLocaleString()}`</h1>
            <h1 className='end-date'><b>End Date: </b>`${new
Date(item.end_time).toLocaleString()}`</h1> <br/>

```

## **5. RESULTS**

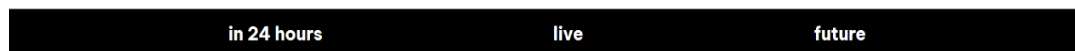
## 5. RESULTS

### 5.1 Coding Contests Detection Website



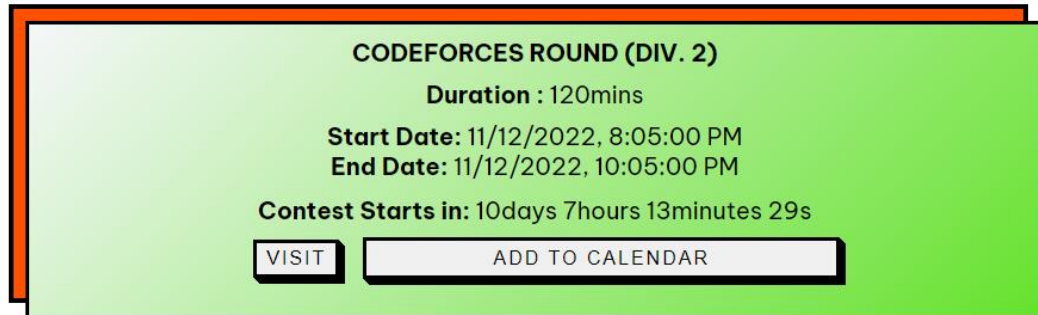
Screenshot 5.1: Coding Contests Detection Website

### 5.2 Three routes live, past, future contests



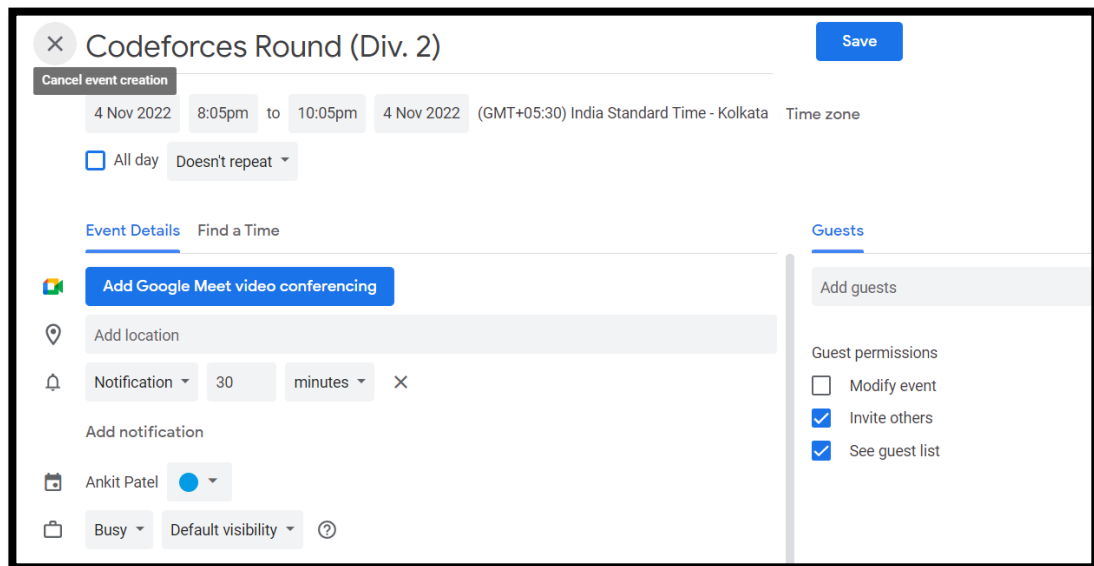
Screenshot 5.2: Three routes live, past, future contests

### 5.3 Individual Contest card



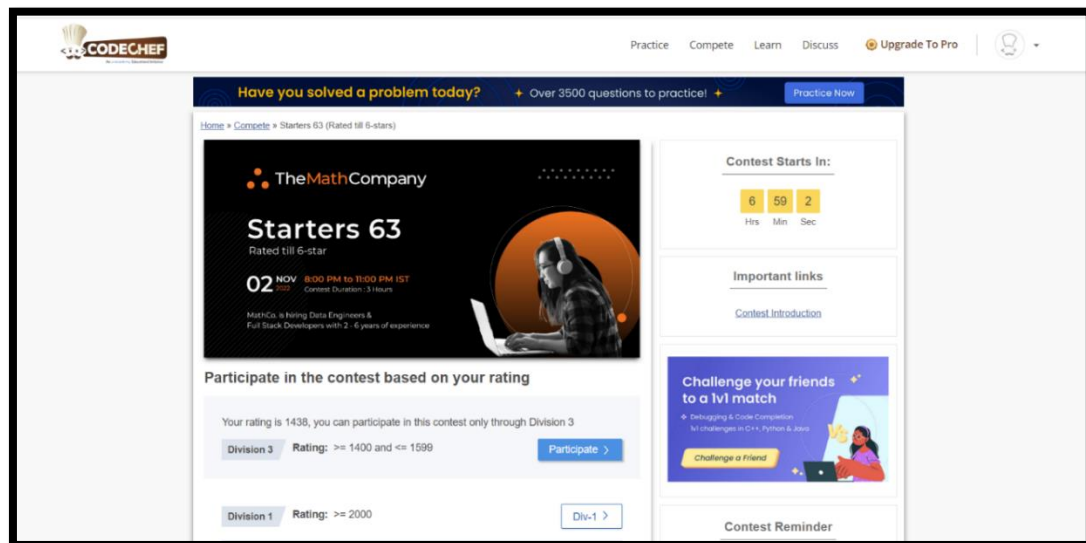
Screenshot 5.3: Individual Contest card

### 5.4 Add to calendar feature preview



Screenshot 5.4: add to calendar feature preview

## 5.5 Contest registration page



Screenshot 5.5: contest registration page

## **6. TESTING**



## **6. TESTING**

### **6.1 INTRODUCTION TO TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

### **6.2 TYPES OF TESTING**

#### **6.2.1 UNIT TESTING**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

## 6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

## 6.2.1 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

## 6.3 TEST CASES

### 6.3.1 CLASSIFICATION

Test caseID	Test case name	Purpose	Input	Output
1	Fetching coding platforms	To detect coding platfor ms.	The user gives the input in the form of a pressing particular platform button.	An output is coding platforms details.
2	Fetching contests details	To detect Contest Details.	The user gives the input in form of selecting particular route.	An output is contests details.

## **7. CONCLUSION**

## **7. CONCLUSION & FUTURE SCOPE**

### **7.1 PROJECT CONCLUSION**

The project CODING CONTESTS DETECTION USING CLIST API main purpose is to help students with their hassle of remembering the coding contest timing of various platforms. Using this website the students can be able to save their time by knowing about all the coding contests in a single place. This project is made to save the time of students who are busy with their schedules and they miss their coding contests due to unawareness of contest timings. We all know the importance of coding contests when the placement is near. This will help them a lot and There will be fewer chances of missing any contest.

### **7.2 FUTURE SCOPE**

In the future, I will add a separate database for each user. An individual user can add all contests he is interested in participating in, and he will get notification of all added contests with a single click. The users will get notifications as soon as any new contests are added to the website.

## **8. BIBLIOGRAPHY**

## **8. BIBLIOGRAPHY**

### **8.1 REFERENCES**

- [1] Sabuj Jana , “A Review Paper on Coding Contests Detection using CLIST Api ,” Int. J. Adv. Res. Comput. Eng. Technol., vol. 1, no. 8, pp. 2278–1323, 2021
- [2] The Road to React: Your journey to master React.js in JavaScript (2022 Edition) by Robin Wieruch.
- [3] Fullstack React: The Complete Guide to ReactJS and Friends by Antony Accomazzo

### **8.2 GITHUB LINK**

[https://github.com/iawesomeankit/coding\\_contest\\_detection](https://github.com/iawesomeankit/coding_contest_detection)