MINI PROJECT -2 REPORT

April (2022-23)

CodeRec

Department of Computer Engineering & Applications

Institute of Engineering and Technology

In partially fulfillment of the award of the degree of

BACHELOR OF ENGINEERING

IN

Computer Engineering and Application

GLA University, Mathura

Submitted By:

Kshitiz Kumar Singh (201500357)

Mohit Varshney (201500409)

Pavan Gupta (201500467)

Utkarsh Srivastava (201500762)

Supervised By:

Mr. Mayank Saxena

(Technical Trainer)

Contents

Declaration

We hereby declare that the work which is being presented in the Mini Project-II **"CodeRec",** in partial fulfillment of the requirement for the Full Stack Project is an authentic record of our own work carried under the supervision of **Mr. Mayank Saxena.**

Kshitiz Kumar Singh (201500357)

Mohit Varshney (201500409)

Pavan Gupta (201500467)

Utkarsh Srivastava (201500762)

Bonafide Certificate

This is to certify that the project entitled **"CodeRec"** carried out in Mini Project-II is a bonafide work done by Kshitiz Kumar Singh**(201500357)**, **Mohit Varshney(201500409), Pavan Gupta(201500467),and Utkarsh Srivastava(201500762)** and is submitted in partial fulfillment of the requirement of the project to be submitted under Mini Project-II for 6th semester.

Signature of HoD:

Head of the Department:

Signature of Supervisor:

Name of Supervisor: Mr. Mayank Saxena

(Technical trainer)

Date:

Acknowledgment

It gives us immense pleasure to present the report of the B.Tech. Mini Project was undertaken during B.Tech. 3rd Year. This project would never have seen the light of day without the help and guidance that we have received.

Our heartiest thanks to **Mr. Mayank Saxena, (Technical Trainer)** for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities toward a constructive goal.

We owe a special debt of gratitude to **Mr. Mayank Saxena**, for his constant support and guidance throughout the course of our work. He has showered us with all their extensively experienced ideas and has also taught us about the latest industry-oriented technologies.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project.

Thanking You All,

Kshitiz Kumar Singh

Mohit Varshney

Pavan Gupta

Utkarsh Srivastava

ABSTRACT

CodeRec is a web application that enables users to compare their coding skills and achievements with other programmers from GitHub, LeetCode, and CodeChef. By entering the usernames or profile URLs of the users they want to compare, CodeRec fetches the user data from each platform and displays it in charts and graphs for easy comparison. CodeRec uses React for the frontend, Node.js and Express for the backend, GitHub API for data retrieval, and Chart.js for data visualization. The motivation for this project was to provide a tool for programmers to benchmark their coding abilities and learn from others. The main challenge was to handle the different APIs and data formats of each platform. The future plans for this project include adding more features like filtering, sorting, and sharing the comparison results.

1. Objective

Coding is a skill that requires constant practice and improvement. Programmers often use online platforms like GitHub, LeetCode, and CodeChef to showcase their coding projects, solve coding challenges, and participate in coding contests. These platforms provide various metrics and statistics to measure the coding skills and achievements of the users, such as number of repositories, commits, stars, followers, problems solved, rating, rank, etc.

However, these platforms have different APIs and data formats, which makes it difficult to compare the user data across them. Moreover, there is no existing tool that allows users to compare their coding skills and achievements with other programmers from multiple platforms in a convenient and comprehensive way.

2. Introduction

2.1 Overview:

CodeRec is a web application that enables users to compare their coding skills and achievements with other programmers from GitHub, LeetCode, and CodeChef. It is designed to help programmers benchmark their coding abilities and learn from others.

The tech stacks used for this project are:

- Frontend: React, HTML, CSS, JavaScript
- Backend: Node.js, Express
- GitHub API: to fetch user data
- Chart.js: to create charts and graphs for the comparison data

The design and architecture of this project are as follows:

- The frontend consists of a single-page application that uses React to create the user interface. The user interface consists of three main components: a form component, a comparison component, and a chart component.
- The form component allows the user to enter the usernames or profile URLs of the users they want to compare. The user can enter up to four users from each platform. The form component

validates the input and sends it to the backend using an HTTP POST request.

- The backend consists of a server that uses Node.js and Express to handle the requests from the frontend. The server uses GitHub API to fetch the user data from GitHub, LeetCode, and CodeChef.

  The server parses and processes the user data and sends it back to the frontend using an HTTP response.
- The comparison component receives the user data from the backend and displays it in a table format for easy comparison. The comparison component shows the metrics and statistics of each user from each platform, such as number of repositories, commits, stars, followers, problems solved, rating, rank, etc.
- The chart component receives the user data from the backend and displays it in charts and graphs for easy visualization. The chart component uses Chart.js to create different types of charts and graphs for the comparison data, such as bar charts, pie charts, line charts, etc.

The challenges and solutions of this project are as follows:

- The main challenge was to handle the different APIs and data formats of each platform. Each platform has its own API

documentation, authentication method, rate limit, data structure, etc. To overcome this challenge, I had to study the API documentation of each platform carefully and use appropriate libraries and modules to fetch and parse the user data.

- Another challenge was to display the user data in charts and graphs for easy visualization. Each platform has different metrics and statistics that may not be directly comparable or compatible with each other. To overcome this challenge, I had to use Chart.js to create different types of charts and graphs for the comparison data and customize them according to the data type and format.

WORKING :

To use CodeRec, the user needs to follow these steps :

- Open the web application in a browser
- Enter the usernames or profile URLs of the users they want to compare in the form component. The user can enter up to four users from each platform.
- Click on the compare button to submit the form
- Wait for the user data to be fetched and displayed in the comparison component and the chart component
- View and analyze the comparison data in the table format and

the charts and graphs

- Use the filter and sort options to refine the comparison data
- Share the comparison results with others using the share button

The expected output of CodeRec is a web page that shows the comparison data of the users from GitHub, LeetCode, and CodeChef in a table format and in charts and graphs. The user can view and analyze the comparison data and learn from others.

3. Requirements

3.1 Software And Hardware Requirements :

- HTML5
- CSS3
- JavaScript
- React
- Node.js
- Express.js
- GitHub API

VSCode :

Visual Studio Code (famously known as **VS Code**) is a free open source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times.

The Visual Studio IDE is a creative launching pad that you can use to edit, debug, and build code, and then publish an app. Over and above the standard editor and debugger that most IDEs provide, Visual Studio includes compilers, code completion tools, graphical designers, and many more features to enhance the software development process.

VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even creating new extensions including code linters, debuggers, and cloud and web development support.

GITHUB :

GitHub is an immense platform for code hosting. It supports version controlling and collaboration and allows developers to work together on projects. It offers both distributed version control and source code management (SCM) functionality of Git. It also facilitates collaboration features such as bug tracking, feature requests, task management for every project.

Essential components of the GitHub are:

- Repositories o Branches o
  Commits o Pull Requests

- Git (the version control tool GitHub is built on)

HTML :

HTML (HyperText Markup Language) is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables. As the title suggests, this article will give you a basic understanding of HTML and its functions.

HTML is a *markup language* that defines the structure of your content. HTML consists of a series of elements, which you use to enclose, or wrap, different parts of the content to make it appear a certain way, or act a certain way. The enclosing tags can make a word or image hyperlink to somewhere else, can italicize words, can make the font bigger or smaller, and so on.

CSS :

CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments.

***JavaScript*:** JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as **LiveScript,** but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name **LiveScript**.

JavaScript is a dialect of the ECMA Script standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions.

JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly. It is a programming language that is characterized as dynamic, weakly typed, prototype-based and multiparadigm.

Advantages of JavaScript:

- ✓ Less server interaction.
- ✓ Immediate feedback to the visitors.
- ✓ Increased interactivity.
- ✓ Richer interfaces.

React : React is an open-source JavaScript library for building user interfaces. It was created by Facebook and is maintained by a large community of developers. React has some key features that make it popular and powerful:

- **JSX**: JSX is a syntax extension that allows you to write HTML-like code inside JavaScript. JSX makes it easier to create and render React elements.
- **Components**: Components are reusable pieces of code that can display data and handle user interactions. Components can be composed together to form complex UIs. Components can also have their own state and props, which are ways of managing data in React.
- **Virtual DOM**: Virtual DOM is a JavaScript object that represents the DOM tree in memory. React uses the virtual DOM to update the real DOM efficiently and only when necessary. This improves the performance and user experience of React apps.
- **One-way data flow**: React follows a one-way data flow model, which means that data flows from parent components to child components through props. This makes it easier to reason about the state of the app and avoid bugs.

Node.js : Node.js is a platform that allows you to run JavaScript code outside the browser. It is based on the Chrome V8 engine, which is a fast and powerful JavaScript engine. Node.js can be used to build various types of applications, such

as web servers, network applications, real-time applications, and desktop applications. Some of the advantages of Node.js are:

- It is fast and efficient, thanks to its event-driven and non-blocking I/O model.
- It is scalable and suitable for high-performance applications, thanks to its ability to handle thousands of concurrent connections.
- It is cross-platform and can run on Windows, Linux, and Mac OS.
- It has a large and active community of developers and a rich ecosystem of modules and frameworks.

Express.js : Express.js is a web framework for Node.js. It is a fast, lightweight and flexible framework that provides a set of features for building web and mobile applications. Express.js has some key features that make it popular and powerful:

- **Middleware**: Middleware are functions that can modify the request and response objects before they reach the final handler. Middleware can be used for logging, authentication, error handling, and more.
- **Routing**: Routing is the process of defining how the application responds to different requests based on the URL and HTTP method. Express.js provides a simple and elegant way of creating routes with parameters, query strings, and wildcards.
- **Template engines**: Template engines are tools that allow you to create dynamic HTML pages using placeholders and logic. Express.js supports various template engines, such as EJS, Pug, Handlebars, and more.

- **Error handling**: Express.js has a built-in error handling mechanism that can catch and handle errors gracefully. You can also define your own custom error handlers for different scenarios.

Github API : GitHub API is a set of RESTful endpoints that allow you to interact with GitHub programmatically. You can use the GitHub API to perform various tasks, such as:

- Manage repositories, issues, pull requests, and other GitHub resources.
- Access user and organization data, such as profile information, activity, and memberships.
- Integrate with GitHub Apps, OAuth Apps, and webhooks to extend GitHub functionality.
- Search for code, users, repositories, and topics across GitHub.

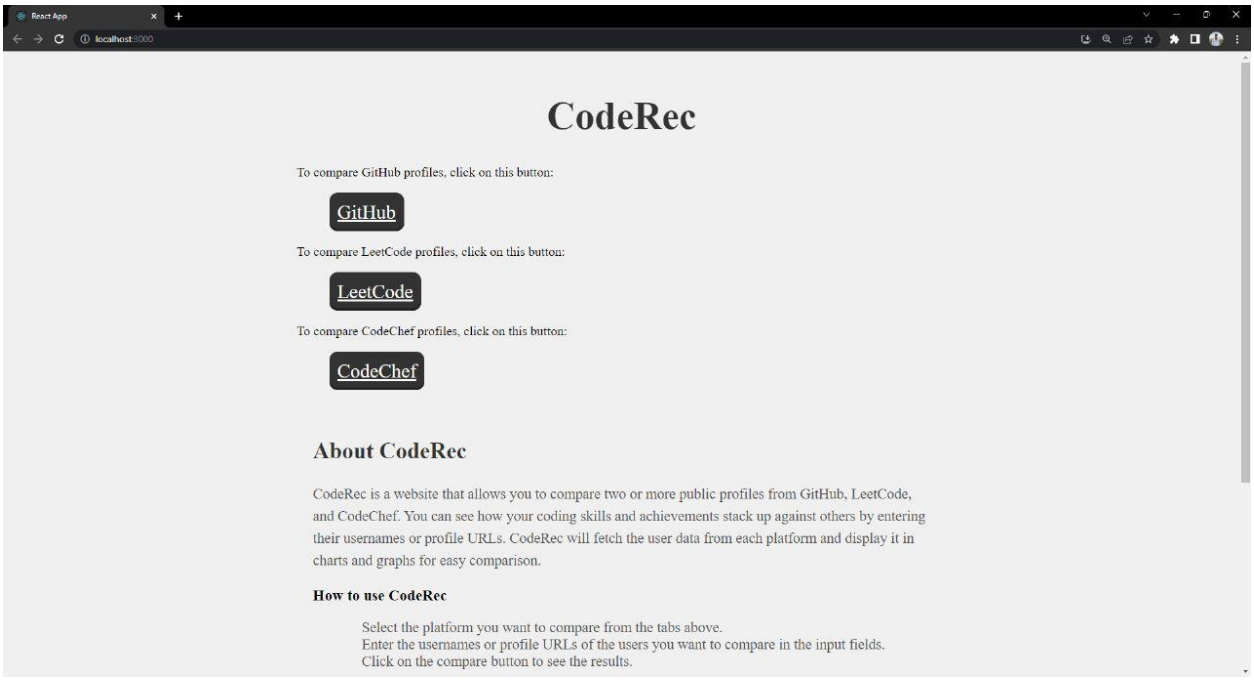
To use the GitHub API, you need to authenticate your requests with a token or a key. You can also use tools like GitHub CLI, curl, or JavaScript to make requests to the API. The GitHub API documentation provides detailed information about each endpoint, including the HTTP method, path, parameters, and examples.

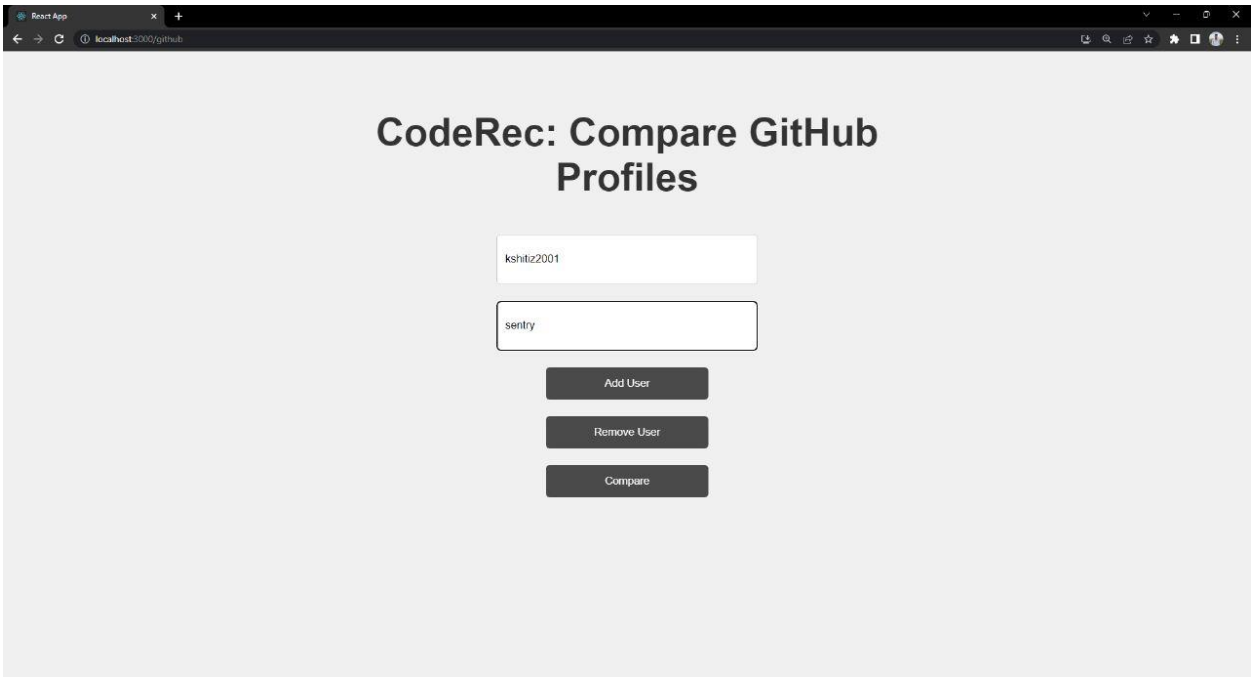
4. Code Repository Link

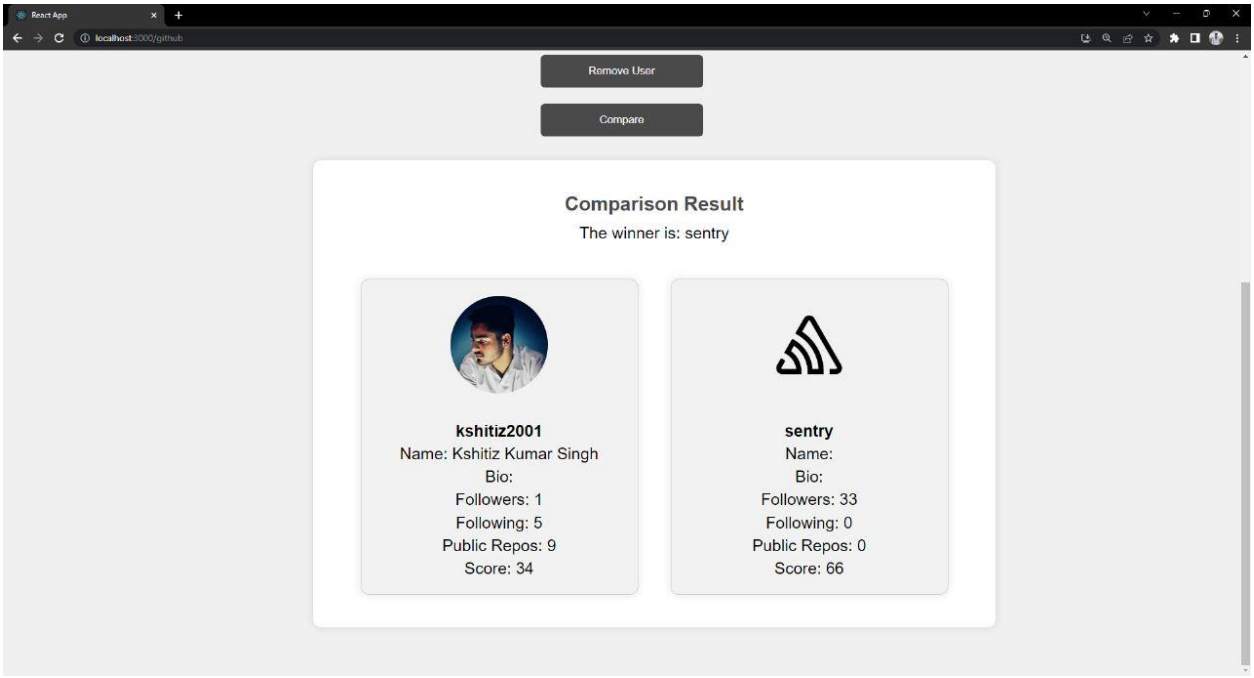
Our source code is available at below repositories you can use the given below link to access the code of our Mini Project-II, **"CodeRec".**

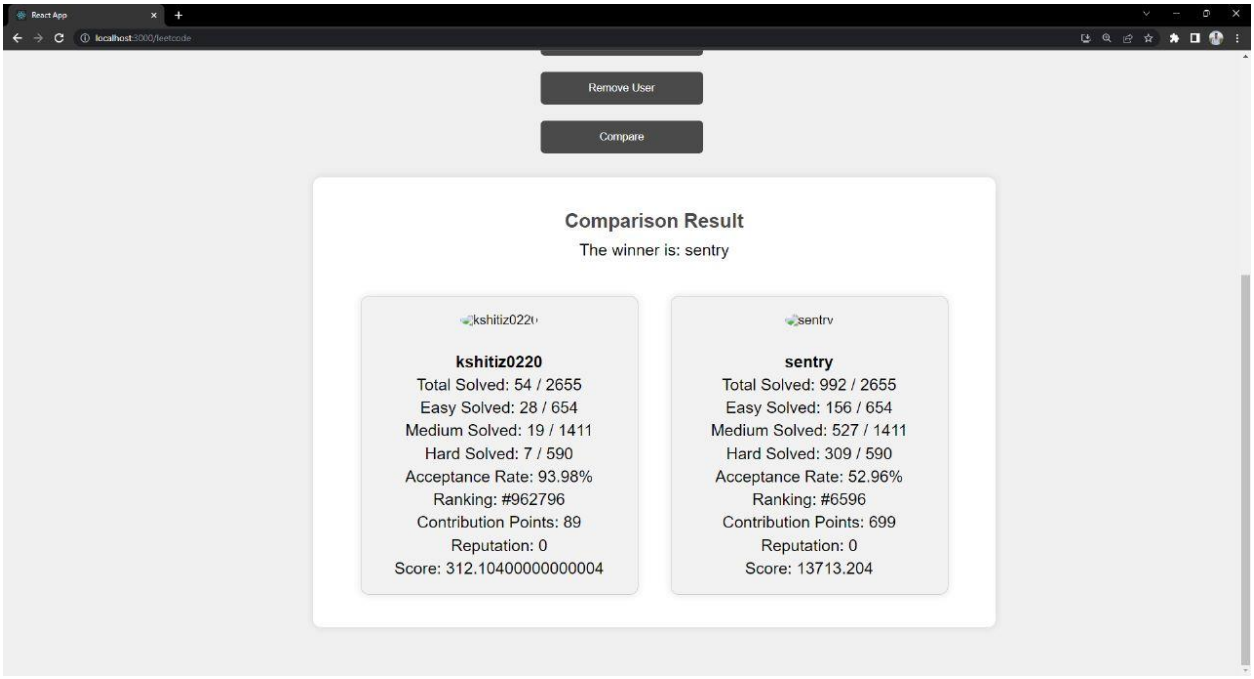
https://github.com/pavangupta352/CodeRec

5. Output Screenshots




React App
localhost:3000/github
CodeRec: Compare GitHub Profiles
kshitiz2001
sentry
Add User
Remove User
Compare


React App
localhost:3000/github
Remove User
Compare
Comparison Result
The winner is: sentry
kshitiz2001
Name: Kshitiz Kumar Singh
Bio:
Followers: 1
Following: 5
Public Repos: 9
Score: 34
sentry
Name:
Bio:
Followers: 33
Following: 0
Public Repos: 0
Score: 66



6. Conclusion

CodeRec is a web application that enables users to compare their coding skills and achievements with other programmers from GitHub, LeetCode, and CodeChef. It is designed to help programmers benchmark their coding abilities and learn from others.

7. References

Books:

- Designing and Web Standards
  -Jeffrey Jeldman with Ethan Marcotte
- React in Action
  -Mark Teileins Thomas (2018)
- Full stack React: The Complete Guide to ReactJS and Friends
  -Nate Murray (2017)

Websites:

- https://developer.mozilla.org/en-US/
- www.google.com
- www.w3schools.com
- https://reactjs.org
- www.projectdeveloper.com
- www.udemy.com/