International Journal of Computer Techniques–IJCT Volume 13 Issue 3, May-June - 2026

Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

"HackConnect": Intelligent Hackathon Collaboration Platform

Tushar Jayant Desai*, Mrunalini Prakash Patil**, Sandhya Arun Tembhurne***, Divakar Shamiron Mukherjee****, Vedant Satish Kothawade*****
Prof.Amarnath Chadchankar******

*(Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email: tushardesai2176@gmail.com

**(Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email:mrunalinipatil707@gmail.com)

*** (Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email:sandhyatembhurne2007@gmail.com)

****(Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email:divakarmukherjee3@gmail.com)

*****(Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email:kothawadevedant07@gmail.com@gmail.com)

******(Department of computer science ,DPGU School of Technology and Research, Pune, India.
Email: Chadchankar.amar@gmail.com

----------**********----------

Abstract:-Hackathons have become one of the most popular ways for students and developers to learn, collaborate, and build real-world projects in a short span of time. These events create an environment where creativity meets pressure, pushing participants to think quickly and work effectively as a team. However, even though hackathons themselves are well organized, the process of preparing for them and participating in them is often messy and unstructured. Participants usually struggle to find hackathons that suit them, connect with the right people, and manage their work efficiently. This is not because of a lack of interest, but because there is no single platform that supports all these needs together in a simple and accessible way.

Most of the existing solutions focus only on a small part of the problem. Some platforms only list hackathons, while others may help with communication or project submission. Because of this, users are forced to switch between multiple platforms, which breaks their focus and

makes the experience unnecessarily complicated. For beginners, this becomes even more difficult, as they may not even know where to start or how to navigate

through all these scattered systems. The lack of proper structure often leads to missed opportunities, poor team formation, and confusion during collaboration.

HackConnect is designed to solve these problems by bringing everything into one place using only web-based technologies. The platform is built entirely using HTML5, CSS3, and modern JavaScript (ES6+), making it lightweight and accessible. All data is managed using the browser's localStorage, which acts as a simple but effective storage system for user accounts, teams, chats, and other features. By using tools like the Google Sign-In API for authentication and the IntersectionObserver API for interactive UI behavior, HackConnect creates a smooth and engaging experience.

The goal is to make hackathon participation simple, organized, and enjoyable, especially for users who are new to the ecosystem.

Keywords:

I. INTRODUCTION

Hackathons have transformed the way people approach learning and innovation, especially in the field of technology. Instead of traditional methods of learning, hackathons provide a space where individuals can directly apply their knowledge, experiment with new ideas, and work collaboratively under time constraints.

This hands-on approach makes hackathons extremely valuable, as participants not only learn technical skills but also develop problem-solving abilities, teamwork, and time management. Despite these advantages, the journey of participating in a hackathon is not always smooth, mainly because the supporting systems are not designed to guide users properly.

One of the most noticeable issues is the difficulty in finding relevant hackathons. Since events are spread across different platforms, users often spend a lot of time searching rather than actually preparing or learning. There is no unified system where users can easily explore opportunities based on their interests or skill level. This creates frustration, especially for beginners who may feel overwhelmed by the lack of direction. Even experienced participants face challenges in keeping track of multiple events and deciding which ones are worth their time .Another major problem lies in team formation. In many cases, participants either join teams randomly or work with people they already know, which limits the diversity of skills within a team. This often results in unbalanced teams where some members are overloaded with work while others are unsure of their role. Without a structured system to match people based on their abilities, it becomes difficult to build effective teams. Communication also becomes an issue, as participants rely on external tools to stay connected, which breaks the workflow and reduces efficiency.

HackConnect addresses these problems by offering a complete browser-based solution that brings all these elements together. By using HTML5 for structure, CSS3 for

styling, and JavaScript for functionality, the platform creates an interactive and responsive experience. Features like localStorage ensure that all user data is stored and managed directly within the browser, removing the need for complex backend systems. With the addition of Google Sign-In API for easy login and modular JavaScript architecture for better

organization, HackConnect provides a smooth and reliable environment for users to participate in hackathons without unnecessary complications.

II. LITERATURE REVIEW

Hackathons have been studied widely as environments that encourage innovation, collaboration, and rapid development. Many discussions around hackathons focus on how they help participants improve their technical skills and build meaningful projects within a short period. They are often described as spaces where learning happens naturally through collaboration rather than structured teaching. However, while hackathons themselves are effective, the tools that support them are often incomplete and do not fully address the needs of participants.

One common observation is that most platforms only handle specific tasks instead of providing a complete solution. For example, some platforms focus only on listing hackathons, while others may provide communication features or project submission tools. This fragmented approach forces users to depend on multiple systems, which can be confusing and inefficient. Instead of having a smooth workflow, participants constantly switch between different tools, which affects their focus and productivity.

Another issue highlighted in discussions about hackathons is the lack of structured team formation. Participants often struggle to find teammates who match their skills and goals. This leads to teams that are not well balanced, which affects both the experience and the outcome of the hackathon.

Communication is also a challenge, as users rely on external platforms instead of having a built-in system that keeps everything in one place[2].

HackConnect builds upon these observations by focusing on simplicity and integration. Instead of introducing complex systems, it uses basic web technologies like HTML5, CSS3, and JavaScript to create a unified platform [3]. Features such as localStorage allow the platform to manage data without needing external systems, while APIs like Google Sign-In and IntersectionObserver enhance usability and interaction. By keeping everything within the browser, HackConnect ensures that users have a consistent and seamless experience.

III. METHODOLOGY

The development of HackConnect follows a simple yet effective approach that focuses on creating a fully functional system using only frontend technologies. The entire platform is built using HTML5, CSS3, and JavaScript, which ensures that it can run directly in the browser without any additional setup. This approach makes the platform lightweight and accessible, allowing users to start using it immediately without worrying about installation or configuration.

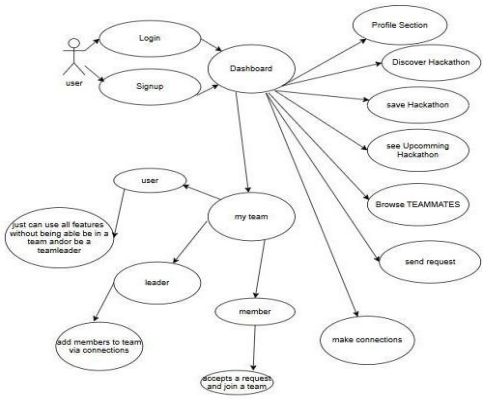
The structure of the platform is based on multiple HTML pages such as index.html, login.html, signup.html, dashboard.html, and profile.html. Each page is designed using semantic HTML5 elements and is

connected through JavaScript, which handles navigation and functionality[3]. CSS3 is used extensively to create a visually appealing interface, with features like Flexbox, Grid, animations, and responsive design ensuring that the platform looks good on all devices.

JavaScript plays the most important role in making the platform interactive. Modern ES6+ features such as arrow functions, async/await, and modules are used to organize the code efficiently. The application follows a modular architecture, where different functionalities are separated into files like utils.js, main.js, and feature-specific modules. This makes the code easier to manage and extend in the future.

Data management is handled entirely through localStorage, which acts as the platform's database. All user information, team data, chat messages, and notifications are stored in structured formats within the browser. This allows the platform to function without a backend while still providing a complete experience. Additional features like Google Sign-In API and IntersectionObserver enhance usability by providing authentication and smooth UI interactions.

Diagram



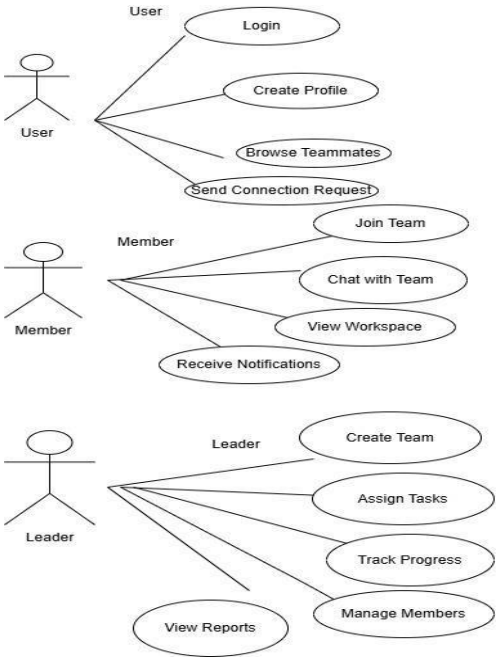


Fig.1 Use Case Diagram of HackConnect System

The use case diagram of the HackConnect system shows how users interact with the platform in a simple and clear way. The main actor in the system is the user, who can perform different activities based on their needs .Users can create accounts either as a member or a leader, log in to the system, and manage their profiles. They can also explore available Hackathons , connect with other users, form teams, and communicate through the chat feature. These actions represent the main functionalities provided by the system .

Overall, the use case diagram helps in understanding how different parts of the system are used and how users interact with it in real-world scenarios.

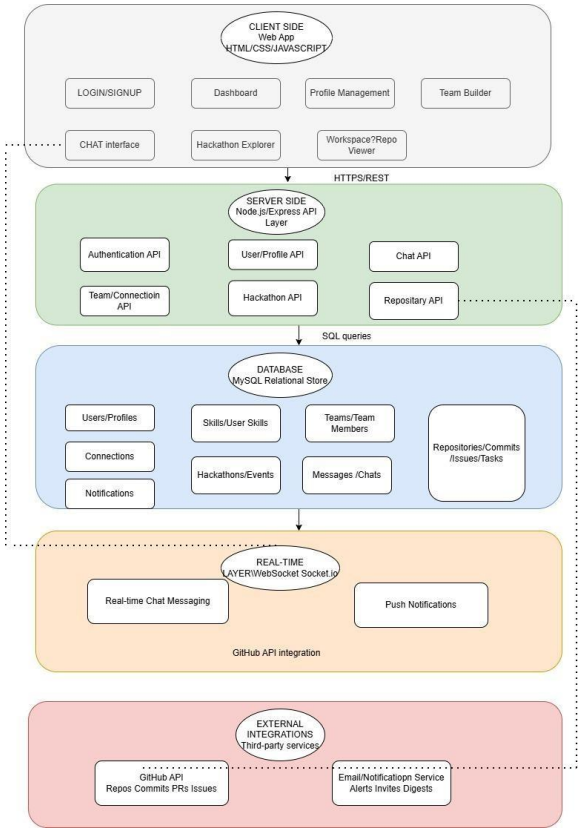


Fig.2 Activity Diagram of Hackconect System

System Architecture

This System follows a modular architecture consisting of :

HackConnect began as a simple frontend application built using HTML, JavaScript, and Local Storage. While this initial version allowed basic functionality, it lacked data persistence, security, and scalability. To overcome these limitations, the system was redesigned into a full-stack architecture.

The upgraded version integrates a FastAPI backend that provides REST APIs and WebSocket support for real-time communication. Security has been significantly improved through JWT-based authentication and bcrypt password hashing.

For efficient data management, SQLAlchemy and Pydantic are used, while

MySQL ensures reliable and persistent storage. Overall, this transformation enables HackConnect to support secure, scalable, and real-time collaboration, making it suitable for more advanced and practical use cases.

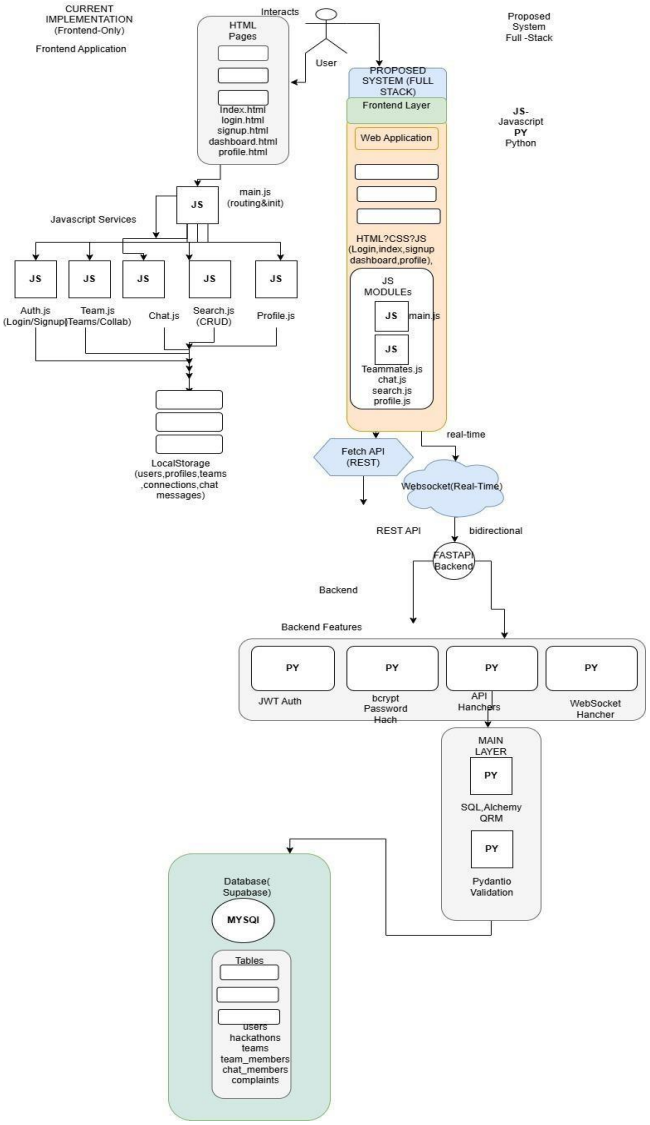


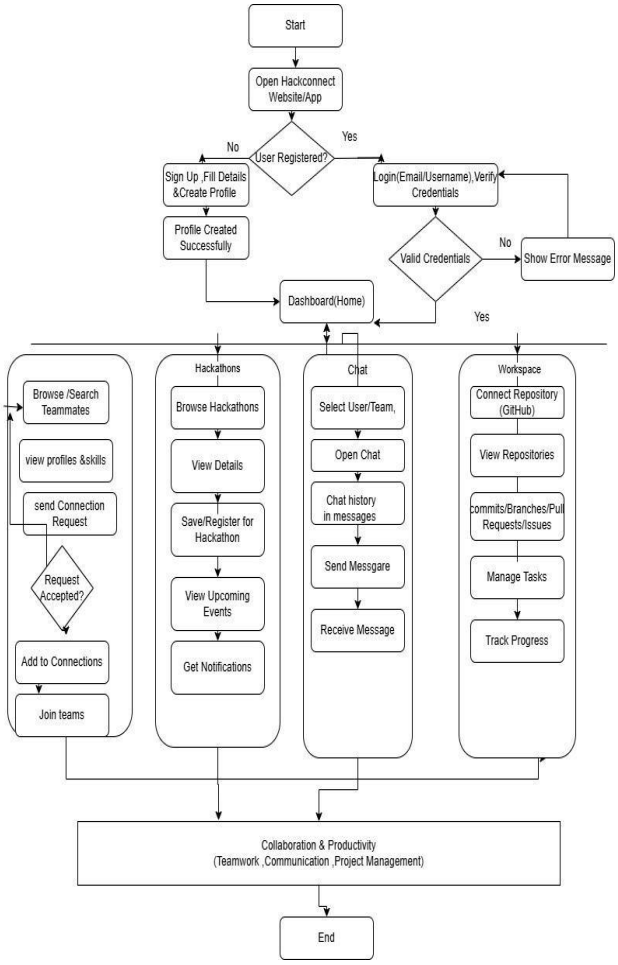
Fig.3 System Architecture of Hacconect System

The HackConnect system flowchart represents the complete user journey from

accessing the platform to collaboration. It begins with the user opening the website or app, followed by a decision step to check whether the user is registered. New users create an account and profile, while existing users log in and verify their credentials.

After successful authentication, users reach the dashboard, which provides access to key modules such as finding teammates, exploring hackathons, chatting with team members, and managing projects through the workspace. These modules enable users to connect, communicate, and collaborate efficiently.

Finally, all activities lead to improved teamwork, communication, and project productivity, completing the system flow.



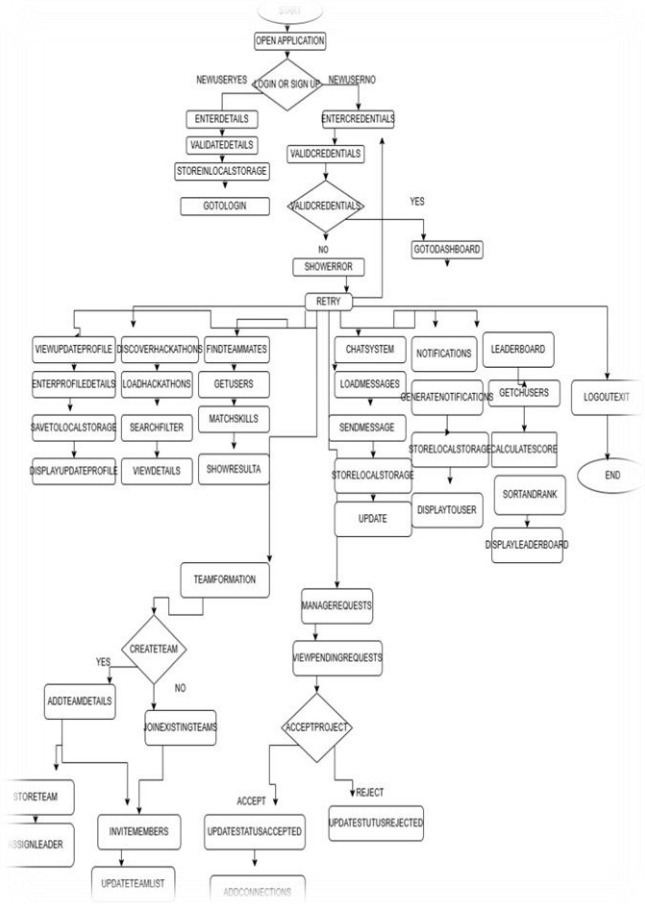


Fig .4 Flowchart for HackConnect System

Algorithm:

Algorithm 1: User Registration & Login
Step 1: Start
Step 2: User opens login/signup page
Step 3: If new user → go to Signup
Step 4: Take input (name, email, password)
Step 5: Validate input fields
Step 6: Store user data in LocalStorage (hackconnect-account)
Step 7: Redirect to Login page

Login Flow:
Algorithm 2: Profile Creation & Update
Step 1: User logs in
Step 2: Open Profile page

Step 3: Input details (skills, bio, interests)
Step 4: Validate input
Step 5: Save/update data in LocalStorage
Step 6: Display updated profile
Step 7: End
Step 8:Users email & password
Step 9: Fetch stored data from LocalStorage
Step 10: If credentials match →

Algorithm 3: Hackathon Discovery

Step 1: User opens dashboard
Step 2: Load hackathon list (static/API/local data)
Step 3: Display all hackathons
Step 4: If user searches/filter → apply filter
Step 5: Show matching hackathons
Step 6: User selects a hackathon
Step 7: Display details
Step 8: End

Algorithm 4: Teammate Search & Matching

Step 1: User opens "Find Teammates"
Step 2: Fetch all users from LocalStorage
Step 3: Compare skills/interests(search bar)
Step 4: Filter matching users(search bar)
Step 5: Display list
Step 6: End

Algorithm 5: Connection Request System

Step 1: Send Request
User selects teammate
Clicks "Send Request"
Create request object:
sender_id = current user
receiver_id = selected user
status = "pending"
Store in LocalStorage (hackconnect-requests)
Show message: Request Sent (Pending)

Step 2: View Pending Requests (Receiver Side)

6. Receiver opens Requests Section
7. Fetch all requests where:
   - receiver_id = current user
   - status = "pending"
8. Display list of pending requests

Step 3: Accept / Reject

9. User selects a request
10. If Accept:
11. Change status → "accepted"
12. Add both users to connections list
13. If Reject:
14. Change status → "rejected"
15. Update LocalStorage

Step 4: Update UI

11. Remove from pending list
12. Show:

- Accepted → in connections
- Rejected → removed / hidden

Algorithm 6: Team Formation

Step 1: User selects "Create Team"
Step 2: Enter team details
Step 3: Save team in LocalStorage
Step 4: Add creator as Leader

Step 5: Invite teammates
Step 6: If user accepts → add to team
Step 7: Update team members list
Step 8: Display team dashboard
Step 9: End

Algorithm 7: Chat System (Frontend-Based)

Step 1: User opens chat
Step 2: Load messages from LocalStorage
Step 3: Display messages
Step 4: User sends message
Step 5: Store message in LocalStorage
Step 6: Update chat UI
Step 7: End

Algorithm 8: Notification System
Step 1: Trigger event (request, invite, message)
Step 2: Create notification object
Step 3: Store in LocalStorage
Step 4: Display in notification panel
Step 5: Mark as read when viewed
Step 6: End

IV. CHALLENGES

One of the biggest challenges in building HackConnect is working without a backend system. Since all data is stored in localStorage, the platform must carefully manage how information is stored, retrieved, and updated. Unlike traditional systems, there is no central database, which means everything has to be handled within the browser itself. This requires careful planning to ensure that data remains consistent and does not get corrupted.

Another challenge is ensuring smooth communication and interaction between different parts of the application. Since the platform uses a modular JavaScript structure, each module must work together without conflicts. Managing this coordination while keeping the code clean and organized is not always easy, especially as the platform grows in complexity.User experience is also a major challenge, as the platform must remain simple and intuitive despite having many features. Designing an interface that is both functional and visually appealing requires careful use of CSS3 features like layout systems, animations, and responsive design. Ensuring that everything works smoothly across different devices and screen sizes adds another layer of complexity.

V. PROPOSED WORK

HackConnect proposes a unified system where all aspects of hackathon participation are handled within a single platform. Instead of relying on multiple tools, users can manage everything directly through their browser. The platform provides features for discovering hackathons, creating profiles, forming teams, and communicating with others, all within a consistent interface.

The system is designed to be simple and accessible, using only web technologies that are widely supported. By avoiding complex backend systems, HackConnect ensures that users can start using the platform immediately without any setup. The use of localStorage allows the platform to store and manage data efficiently, while JavaScript handles all interactions and updates in real time.

The modular architecture of the platform makes it easy to extend and improve in the future. Each feature is implemented as a separate module, allowing developers to add new functionalities without affecting existing ones. This approach ensures that the platform remains flexible and scalable as it evolves.

VI. RESULTS AND DISCUSSION

The implementation of HackConnect shows that it is possible to create a complete hackathon platform using only frontend technologies. Users are able to perform all essential tasks, from creating accounts to forming teams and communicating with others, without needing any external systems. This demonstrates the effectiveness of using localStorage as a lightweight data management solution.

The platform also provides a smooth and interactive user experience, thanks to the use of modern CSS3 features and JavaScript.

Animations triggered by IntersectionObserver create a dynamic interface, while responsive design ensures that the platform works well on different devices. These elements contribute to a more engaging and enjoyable experience for users.

However, there are also limitations, particularly related to data storage and scalability. Since all data is stored locally, it is limited to the user's browser and cannot be shared across devices. Despite this, the platform serves as a strong proof of concept, showing how a fully functional system can be built using only basic web technologies.

VII. CONCLUSION AND FUTURE SCOPE

HackConnect successfully demonstrates how a unified hackathon platform can be created using only frontend technologies. By combining HTML5, CSS3, and JavaScript with tools like localStorage and Google Sign-In API, the platform provides a complete solution for managing hackathon participation. It simplifies the process, making it easier for users to focus on learning and collaboration rather than dealing with multiple tools.

The platform is especially helpful for beginners, as it provides a clear and organized structure that guides them through the process. Features like modular architecture and responsive design ensure that the platform remains flexible and user-friendly. Even though it does not use a backend, it still manages to deliver a meaningful and functional experience.

In the future, HackConnect can be further improved by enhancing its features and refining its user interface. Additional modules can be added to provide more functionality, while existing features can be optimized for better performance. The platform has strong potential to grow into a powerful tool for hackathon participants, proving that even simple technologies can create impactful solutions when used effectively.

WebSockets, secure authentication using JWT, and database integration (MySQL/Supabase) can be implemented to improve performance and security. The platform can also include AI-based recommendations to suggest suitable teammates and relevant hackathons based on user skills and interests.

VIII. REFERENCE

[1] **Devpost Platform — A leading platform for hackathon discovery, but with limited team collaboration or AI matchmaking.**
link https://devpost.com

[2] **Design Thinking in Hackathons** — Highlights common practices successful teams use; motivates HackConnect's team tools like tasks & readiness score.
Link;-https://arxiv.org/abs/2206.04744
(Also published in journal form: https://doi.org/10.1080/10447318.2022.2075601)

[4] **Platform** :- **This paper is published in the *International Journal for Research Trends and Innovation (IJRTI), Volume 10, Issue 4 (2025).***
https://www.ijrti.org/papers/IJRTI2504280.pdf

[5] App :-
Hachmate:-
Reference to our project.

The proposed system is inspired by the Hackmate application, which focuses on connecting individuals for collaboration and teamwork. Similarly, HackConnect aims to bring users together on a single platform, with a focus on hackathons, project building, and team formation. Based on this idea, HackConnect extends the concept by including:

Hackathon-specific team creation

Role-based access (User, Member, Leader)

Project collaboration features

Event participation management

HackConnect enhances the core idea of Hackmate by adapting it specifically for hackathon environments and improving collaboration and team management.

[6] **Website :-**

Matchminds:-" A Hackathon Teammate Recommendation Website link", Available at:
GitHub:- https://github.com/harshitachhangani/MatchMinds.