

# Chapter 3

## 3.1 What is Fest Hub

Fest Hub is a comprehensive web-based event management system designed to streamline the planning, organization, and execution of college technical and cultural fests. Traditionally, event management in colleges relies on manual processes, spreadsheets, and fragmented communication through emails and messaging apps. Some institutions use generic event management software, but these solutions are often not tailored to academic fests, leading to inefficiencies in participant registration, scheduling conflicts, and difficulty in result authentication. Moreover, existing systems lack transparency in judge allocation and result processing, making it challenging to ensure fairness. The absence of a centralized and automated platform results in increased administrative workload, miscommunication, and difficulties in tracking event progress.

To address these challenges, Fest Hub provides a centralized platform for event scheduling, participant registration, judge allocation, and real-time communication between students, faculty, organizers, and external participants.

Fest Hub offers several advantages that enhance the efficiency and reliability of college fest management. One of its key features is blockchain integration for secure results, which ensures result authenticity, prevents tampering, and builds trust in the evaluation process. The system provides a user-friendly interface, making event management simpler and more accessible for organizers, participants, and judges. Additionally, Fest Hub incorporates scheduling and coordination, eliminating scheduling conflicts and optimizing event planning. It also facilitates seamless communication by delivering instant updates, notifications, and announcements to all stakeholders, ensuring smooth coordination. Moreover, Fest Hub includes a dedicated mobile application, allowing users to register, receive real-time event updates, notifications, track schedules, and access result from anywhere. This mobile accessibility enhances convenience and engagement, making Fest Hub a comprehensive and efficient solution for managing college technical and cultural fests.

A key feature of Fest Hub is its integration of blockchain technology for secure result declaration. This ensures data integrity, prevents result manipulation, and enhances transparency. By offering a user-friendly interface, Fest Hub reduces administrative burdens, enhances accessibility, and promotes efficient management of large-scale events involving multiple institutions.

### **3.2 Approaches of Fest Hub**

Fest Hub follows the Model-View-Controller (MVC) architecture, a widely used design pattern that separates application logic into three interconnected components: Model, View, and Controller. This approach enhances modularity, scalability, and maintainability, making Fest Hub an efficient and well-structured system.

- **Model (Data Layer):** The Model represents the data and business logic of Fest Hub. It manages database interactions, including user registrations, event details, judge allocations, result storage. The integration of blockchain technology for secure result declaration and certificate authentication is handled at this layer, ensuring data integrity and transparency.
- **View (Presentation Layer):** The View is responsible for the user interface, displaying event details, schedules, participant lists, and results. It provides an interactive and user-friendly experience for students, faculty, organizers, and judges. The web-based interface, along with a dedicated mobile application, ensures accessibility across devices, allowing seamless navigation and real-time updates.
- **Controller (Logic Layer):** The Controller acts as the intermediary between the Model and the View. It processes user requests, validates inputs, retrieves data from the Model, and updates the View accordingly. For instance, when a participant registers for an event, the Controller validates the input, stores the data in the database, and updates the event dashboard. It also handles authentication, role-based access, and communication between different system components.

### **Blockchain Technology:**

Blockchain is a decentralized and distributed ledger technology that records transactions across multiple nodes in a secure, transparent, and immutable manner. Each transaction is stored in a "block," which is linked to the previous block, forming a continuous "chain." This structure ensures

data integrity, making it nearly impossible to alter or manipulate stored information without consensus from the network. Key features of blockchain include:

- Decentralization: No single entity controls the data, reducing the risk of manipulation.
- Immutability: Once a block is added to the chain, it cannot be altered, ensuring data integrity.
- Transparency: Transactions are recorded in a verifiable and auditable manner.
- Security: Cryptographic hashing and consensus mechanisms (e.g., Proof of Work or Proof of Stake) prevent unauthorized changes.

Fest Hub leverages blockchain technology primarily for secure result declaration ensuring transparency, authenticity, and fraud prevention in college fest management. The integration process involves the following steps:

### **1. Choosing a Blockchain Platform**

Smart contracts can be deployed using Ethereum-based Solidity programming.

### **2. Implementing Smart Contracts**

- a. Smart contracts are self-executing contracts with predefined rules stored on the blockchain.
- b. A smart contract is created for managing event results, verifying judge inputs.

### **3. Blockchain-Based Result Declaration**

- Once judges submit event results, they are hashed and stored on the blockchain.
- The result hash ensures data authenticity and prevents tampering.
- Participants and organizers can verify results using blockchain transactions.

## **3.3 How Does Fest Hub Works**

The Fest Hub system operates as a centralized platform that seamlessly connects various stakeholders involved in a college fest, ensuring efficient event coordination, transparent result declaration, and real-time communication. The system is role-based, with different users—Admins,

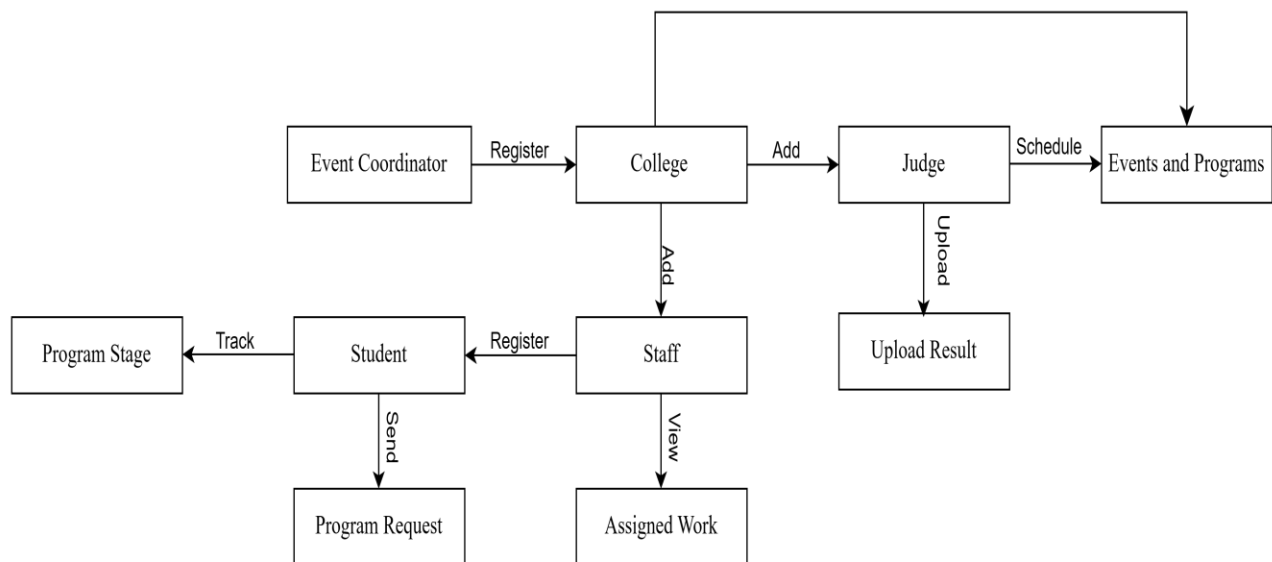
Event Coordinators, Colleges, Staff, Judges, and Students—each having specific functionalities to manage and participate in the event.

The process begins with the Event Coordinator taking charge of organizing the fest by registering colleges, adding judges, and scheduling events and programs. The coordinator ensures that all necessary information is updated in the system for smooth execution.

Colleges participating in the fest have their own portal to manage their profile, add staff members, and view scheduled events and programs. Staff members, assigned by colleges, are responsible for managing student registrations and viewing assigned tasks. They can also track their work and update student details as required. Judges, on the other hand, have access to their assigned events, where they evaluate performances and upload competition results directly into the system. To ensure fairness and security, the results are stored immutably using blockchain technology, preventing any manipulation or unauthorized changes.

Students, as primary participants, log into the system to browse events, check schedules, and send requests to participate in specific programs. Before their scheduled events, they receive real-time notifications ensuring they are well-prepared and track the stage or venue where the program is being conducted. After the event, students can view their competition results, check their accumulated points. Additionally, they can provide feedback about their experience, helping improve future fest management. The admin, who logs in securely to oversee and manage the fest. The admin verifies and approves Event Coordinators, views ongoing programs, and monitors event details, staff assignments, and judges.

The entire workflow ensures a structured, efficient, and transparent fest management system, reducing manual errors and enhancing user experience. With blockchain integration, the system maintains credibility in result declaration, while automated notifications and a well-organized user interface streamline participation and coordination. Through Fest Hub, the fest is not only easier to manage but also more engaging and accessible for all stakeholders.



### 3.4 Module Description

System modules are individual components or sections of a software system, each designed to perform specific functions within the larger application. By breaking down a system into these smaller, manageable units, developers can build, test, and modify functionalities independently, which enhances modularity, scalability, and maintainability. Modules usually communicate with each other, passing information and coordinating to deliver a seamless user experience.

#### 3.4.1 ADMIN

- Login: Admin can securely log into the system
- Verify: Approve Event Coordinator
- View: View all programs added by event coordinator
- View: View all Events, Staff details and Judge details

#### 3.4.2 EVENT COORDINATOR

- Login: Coordinator can securely log in to the system.
- Add Colleges: Add or verify registered colleges to participate in the fest.
- Add Judge: Add or View Judge details
- Add Event & Program: Add Details of Events and Programs

### 3.4.3 COLLEGE

- Manage: Colleges can login and manage profile.
- Add: Add Staff Details to assign the works.
- View Event: View Event Details provided by the coordinator
- View Program: View Program Details of the event & Result

### 3.4.4 STAFF

- Manage: Staff can login and Manage Profile
- Add & View: Add & View Student Details who are participating the program
- View Work: View Assigned Work by College

### 3.4.5 JUDGE

- Manage: Judge can login and Manage Profile
- View: View Assigned Work by event coordinator
- Upload: Upload Result of the program

### 3.4.6 STUDENT

- Login: Students can log in using their credentials and Manage Profile.
- View Event & Program: View Events & Check the schedule of programs.
- Send Request: Send Request for Program.
- Receive Notification: Receive notification before 5 minutes of their program.
- View Result: Access their performance results for the programs.
- Send Reviews: Submit feedback or reviews about events or their experience.
- Point Table: View the points scored by themselves or their college.
- Track: Track Stage where the program is conducted.

## 3.5 Software Tools

**Table:** Software tools or languages used for the project development

Operating System	Windows 10/11
Front End	JavaScript, HTML, CSS, Bootstrap

Back End	Python, Flutter
Framework	Django, Flutter
Database	MySQL
IDE	PyCharm
Version Control	Git

### 3.5.1 Python

Python serves as an ideal backend language for web development, thanks to its simplicity, versatility, and extensive range of libraries. As a high-level programming language, Python allows developers to write code that is readable and maintainable, reducing the complexity typically associated with backend development. Its support for a variety of libraries and frameworks enables efficient handling of essential backend functions such as user authentication, data processing, and API integrations, making it an excellent choice for web applications of any scale. Python's scalability also ensures that as an application grows in complexity and user demand, it can handle larger data volumes and more intricate backend logic without significant performance issues.

### 3.5.2 Django

Django is a powerful, Python-based web framework designed to simplify the backend development process. It follows a "batteries-included" philosophy, offering built-in tools for many common tasks like database migrations, form processing, and URL routing. Django's Model-View-Template (MVT) architecture encourages structured, modular development by separating data handling, business logic, and presentation layers. This structure helps in building scalable applications and maintaining code that is both organized and easy to extend. With Django, developers can create powerful applications faster, adhering to the framework's "Don't Repeat Yourself" (DRY) principle, which encourages code reuse and eliminates redundancy.

Django also includes robust security features to protect against threats like SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF), making it a trusted choice for secure web application development. Its community and extensive documentation further enhance its usability, allowing developers to access ready-made solutions for common issues and receive

support for best practices. Overall, Django simplifies backend development in Python, allowing for the rapid development of secure, scalable applications.

### 3.5.2 MySQL

MySQL is a widely used, open-source relational database management system (RDBMS) that is known for its reliability, performance, and ease of use. It is a popular choice for web applications due to its efficient handling of large datasets, making it suitable for applications with high transaction volumes or complex data structures.

MySQL organizes data into structured tables, which can be queried and managed using SQL (Structured Query Language). With features like indexing, data integrity checks, and transaction support, MySQL ensures data is stored securely and can be accessed quickly. Its scalability allows it to handle both small-scale applications and enterprise-level deployments, making it highly versatile across industries.

MySQL is also compatible with various platforms, including Windows, Linux, and macOS, and integrates seamlessly with many programming languages and frameworks. Additionally, its large community and extensive documentation provide developers with resources for troubleshooting, optimization, and implementing advanced database configurations, making MySQL a robust and accessible solution for data management in a wide range of projects.

### 3.5.4 Flutter

Flutter is a cross-platform framework that allows us to build a mobile application with a single codebase. It ensures a smooth and interactive user interface, making it ideal for our project's front-end development. With its rich widget library and fast rendering engine, Flutter provides a seamless user experience while maintaining high performance across different devices.

### 3.5.5 Git

Git is an excellent choice for version control due to its efficiency and reliability in managing code changes. It supports seamless collaboration by enabling branching and merging, preventing code conflicts. Its distributed nature allows each contributor to track changes locally and push them to a shared repository when ready, ensuring data integrity and easy version rollback. Platforms like



GitHub and GitLab enhance Git by offering centralized repositories, issue tracking, and continuous integration tools, making collaboration and project management more efficient.

#### 3.5.6 PyCharm

PyCharm is a powerful integrated development environment (IDE) that supports Python, Flutter, and web development. It provides advanced debugging tools, intelligent code suggestions, and seamless integration with Git, enhancing productivity. Its user-friendly interface and built-in package manager make development more efficient and streamlined.

#### 3.5.7 Solidity

Solidity is a high-level, object-oriented programming language designed for writing smart contracts that run on Ethereum and other EVM (Ethereum Virtual Machine) compatible blockchains. It is influenced by JavaScript, Python, and C++, making it easy to learn for developers familiar with these languages.