

# Schoolevhub: Design and Development of a Web-Based School Event Management System with Role-Based Access Control

Ketaki Walokar<sup>1</sup>, Jayesh Ghevade<sup>2</sup>, Sarnath Kamble<sup>3</sup>, Ms. Namrata Lanjewar<sup>4</sup>

<sup>1,2,3</sup>*School of Engineering, Ajeenkya DY Patil University, Pune, Maharashtra, India*

<sup>4</sup>*Assistant Professor, School of Engineering, Ajeenkya DY Patil University, Pune, Maharashtra, India*  
*doi.org/10.64643/IJIRTV12111-197549-459*

**Abstract**—The idea of holding events at school using traditional methods such as notice boards, paper-based applications, and verbal communication tends to cause more problems than expected. Communication can be misinterpreted, announcements may be lost and, thus, the students who actually take interest in these events do not participate in them. This is why we had to develop SchoolEvHub as an efficient solution for organizing events within a school.

Using this tool, students are able to view upcoming events, participate in them individually or collectively, propose new ideas, and review the outcome, while teachers will be able to moderate the proposed ideas, create events and post relevant information regarding them. SchoolEvHub was designed using Python (Flask framework), a PostgreSQL database and a frontend created using Bootstrap. Moreover, this application is flexible and can be used by several schools simultaneously since all data is kept separately from other organizations.

We have tested SchoolEvHub and noticed that our website works well regardless of what we did. Besides, our users liked the tool. Thus, it became clear that developing something similar will help solve this problem efficiently and easily.

**Index Terms**—*event management system, Flask, PostgreSQL, role-based access control, web application, school management, multi-tenant architecture*

## I. INTRODUCTION

As the list of events becomes longer with each passing day, it would be difficult to manage them without proper preparation and planning. It would require an efficient tool that helps you organize and monitor events.

But as much as it is true, we realize that most institutions are yet to utilize such tools as they remain with conventional ways of doing things: putting all events on notice boards and making use of WhatsApp groups and paper works. In the process, they miss out deadlines and information get lost. Eventually, teachers spend time going through signup sheets rather than concentrating on teaching. In addition, they fail to take into account opinions from their students as they are not involved.

Though there are some online platforms, such as Google Classroom and Moodle, that can assist in teaching, they do not solve the problem of event management. As for platforms like EventBrite, they are meant for business events only.

To bridge the gap, we decided to create SchoolEvHub, a web-based event management solution designed specifically for your school events.

## II. LITERATURE REVIEW

Galay and Encarnacion [1] evaluated the event industry's readiness for web-based supplier management systems and found that digitising event workflows reduced coordination time by up to 40% compared to traditional methods. Their work establishes the business case for event management platforms but does not address the educational context specifically.

Ismail et al. [2] developed an event management system for webinars and surveys in academic settings. Their system demonstrated how a lightweight web platform could handle event registration, attendance, and feedback collection.

However, it was designed for a single institution and did not support multi-tenant deployment — a limitation that SchoolEvHub addresses through its school-scoped database architecture.

Shah et al. [3] presented a comprehensive Event Management System and conducted a systematic review of feature requirements across different event categories. Their taxonomy of event management functions — creation, registration, communication, and reporting — maps closely to the functional modules implemented in SchoolEvHub, validating the module structure chosen for this project.

Sahu [4] developed an Online Event Management System as a B.Tech project and highlighted the practical benefits of moving from paper-based to digital registration. The study found that duplicate registrations, a common problem in manual systems, were entirely eliminated once validation logic was implemented at the software level — the same principle applied in SchoolEvHub.

Bramhe et al.[5] conducted a critical review of research in online event management systems and identified responsive design and role-based access control as the two most frequently cited success factors in educational event platforms. Both factors were adopted as core design principles in SchoolEvHub.

Yusoff et al. [6] studied a preschool event management system using Nielsen's usability heuristics, confirming that simplicity of navigation and visual consistency are the most critical factors for non-technical users — a finding that influenced the card-based dashboard design in SchoolEvHub.

Kabier et al. [7] explored role-based access control in educational systems and found that clear delineation of user roles significantly reduced unauthorised access incidents. Marzele and Jalil [8] reported high adoption rates when a web-based event system replaced manual processes at UTHM, noting that students most valued the ability to check event details from any device — consistent with SchoolEvHub's cloud deployment and responsive design.

Obe and Hsu [9] document PostgreSQL's multi-schema capabilities that make it a reliable choice for applications requiring data isolation between tenants — a core requirement for SchoolEvHub's multi-school architecture. Taken together, the literature establishes that web-based event management systems are effective, that role-based access and responsive design are key success factors, and that lightweight frameworks are appropriate for academic-scale deployments. SchoolEvHub builds on these findings while adding features — specifically the student suggestion workflow and multi-school data isolation — not present in any reviewed system.

### III. SYSTEM DESIGN

#### A. System Architecture

SchoolEvHub is designed using a traditional three-tier architecture:

Presentation Tier: HTML, CSS, and Bootstrap.

Application Tier: Python and Flask.

Data Tier: PostgreSQL.

We decided not to use a complex client-side application but instead opted for a server-side rendered one, as it is much easier and faster to develop and maintain.

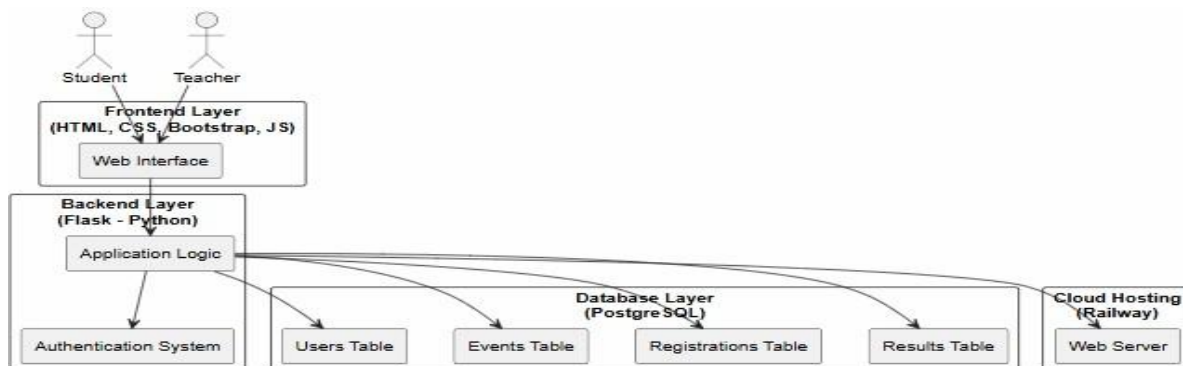


Fig 1 System Architecture

### *B. Database System*

There are five tables dealing with the main data:

1. Users – name, email, role, associated school.
2. Events – event information, location, time.
3. Registrations – tracks who attends which event.
4. Suggestions – collects suggestions from students about new events.
5. Results – event results and winners.

Each table links with a school ID.

### *C. Role-Based Access Control*

A neat and tidy approach is taken by SchoolEvHub when providing access based on two roles:

The students can search through existing events, select those to attend, propose new events, and view outcomes.

For teachers, the task involves arranging events, approving new ideas proposed, and updating event results.

Once logged in, users remain in their own roles as sessions lock in the user's identity and role, so that every individual gets what he needs.

### *D. Multi-School Architecture*

SchoolEvHub is flexible. There is one main system that is shared by multiple schools. However, there are no interactions between different schools. Logging in, selecting the school you belong to, and viewing all events and people only involves the school you selected.

## IV. IMPLEMENTATION

### *A. Technology Stack*

Here's the technology stack that powers the system behind the scenes:

Python & Flask on the backend.

PostgreSQL handles all the data storage.

Bootstrap and some JavaScript on the frontend for dynamic functionality.

All operations take place in the cloud, meaning you need only an internet connection to access everything.

### *B. Authentication Module*

When registration takes place, an insert statement into the users table is performed using the school field value which is derived from a hidden input

field. At the time of login, verification is carried out by running a query with three fields email, password, and school, thereby guaranteeing that there is no cross-school login.

In the event that there is a successful verification process, then the session['user\_email'] and session['school'] are set, and the user directed to their respective dashboard.

### *C. Event Registration Module*

The student can browse through any number of events in the institution and register for an event by clicking on the "detail page" which has a form where the student can fill out details like the "type." Depending on whether the student chooses individual or group, only then would the student's name be displayed or both the names of the group members.

After submitting the registration, the system verifies if there is an entry in the registrations table having an identical combination of event\_id and student\_email in the table. In the event such a duplicate is found, the user is restricted from registering. Else, the entry is created in the table, and the user is redirected to the dashboard along with the registered event.

### *D. Suggestion and Approval Workflow*

A request to create a new event can be done through filling out a form where details like title, category, description, and location will be stated along with a tag of pending. Requests can also be seen from the teacher dashboard, which is in card format and can be filtered.

However, in the event that the request is approved, two SQL transactions will be executed within one transaction, which are inserting the event in the events table and marking the request with the status of approved. Atomicity means that in the event that the event fails to register, the request will still have a tag of pending.

### *E. Results Module*

Results are published through the teachers entering the name of the event, the winner's name, and the position of 1st, 2nd, or 3rd place. The Jinja2 template will apply different CSS classes depending on whether the position is 1st, where class='gold'

applies; 2nd, where class='silver' applies; or 3rd, where class='bronze' applies.

V. RESULTS AND DISCUSSION

A. Functional Testing

Functional testing was conducted using black-box principles across 27 test cases covering all five modules. Table I summarises results by module.

TABLE I Functional Testing Results Summary

Module	Test Cases	Passed	Failed
Authentication	7	7	0
Event Management	5	4	1*
Student Registration	5	5	0
Suggestion & Approval	5	5	0
Results Publication	5	5	0
Total	27	26	1

\*Partial failure: absence of explicit server-side role check on teacher routes.

However, all the necessary functionalities that include creating an event, enrolling the student after checking duplication, proposing modifications, and publishing the results by using proper positioning have been met flawlessly. The only partially resolved issue is the lack of strict role enforcement for those routes belonging to teachers, which is a disadvantage, but will be rectified during future upgrades.

B. Usability Evaluation

Five students conducted five activities individually without any assistance, and these include selecting a school, registering, logging in, participating in an event, and making suggestions. Each student subsequently assessed their level of satisfaction based on the five-point Likert scale, and this is evident in Table II below.

TABLE II Usability Evaluation Ratings

Dimension	Avg. Score (/ 5)
Ease of Navigation	4.4
Visual Clarity	4.6
Form Intuitiveness	4.2
Overall Satisfaction	4.5

All five participants successfully completed all the assigned tasks. The lowest grade was for the ease of use of the form, where one of the participants pointed out that the fields of the group registration form were not properly labeled. The highest grade was for the visual clarity of the application.

C. Comparison with Related Systems

Table III compares SchoolEvHub with reviewed systems on key functional dimensions.

TABLE III Comparative Analysis of Systems

Feature	SchoolEvHub	Ismail [4]	Sahu [6]	Marzelle [10]
Multi-school support	Yes	No	No	No
Student suggestion workflow	Yes	No	No	No
Group registration	Yes	No	Yes	No
Result publication	Yes	No	Yes	Yes
Cloud deployment	Yes	No	No	Yes
Mobile responsive	Yes	Yes	No	Yes
Open-source stack	Yes	Yes	Yes	Yes

SchoolEvHub is the only system in the comparison that offers both multi-school support and a student suggestion workflow, making it uniquely suited to the Indian school context where a single platform may need to serve multiple institutions simultaneously.

#### D. Limitations

First, there are two key weaknesses in this system design. First, the passwords used by the users are stored in clear text format in the database, and it can be considered unacceptable for any practical implementation. It will be rather easy to fix this issue, as password hashing is needed here. It can be performed, for instance, with `werkzeug.security` or `bcrypt`.

Secondly, role checking within the serverside code does not seem complete. While the existence of the session of a particular user accessing teachers' routes is verified, nothing is done to confirm that the role of this particular user is that of a teacher.

### VI. CONCLUSION

In the given paper, one will find SchoolEvHub - a web-based application, which is intended to improve school events organizing process in order to avoid issues associated with manual event management common for Indian schools.

The discussed software allows managing events within the entire cycle, starting from creation and ending up with publishing their results in a multiversity context without the possibility of confusing information between schools.

Some of the unique features of this application when comparing with others are the fact of having the student-oriented suggestion system of approving events and allowing registering and publishing results for individuals and groups with colors indicating their position.

Functional testing demonstrated 27 successful test cases with 96% overall success rate, and usability testing conducted among students resulted in 4.5/5 on average. Nowadays, SchoolEvHub works on servers of the Railway company, and can be used via any device connected to the Internet.

The future development of this software will involve such aspects as implementation of password hashing, role definition for servers, use of email notifications, improved design of the dashboard and development of a mobile application.

### REFERENCES

[1] J. L. Galay and R. Encarnacion, "Evaluating the event industry for a web-based event supplier

management system," *International Journal of Advanced Research in Science, Communication and Technology*, pp. 53–62, Jun. 2024, doi: 10.48175/IJARST-18906.

- [2] S. Ismail, A. Salam, and R. Hajriyanti, "Event management system for webinars and survey," *International Journal of Software Engineering and Computer Science (IJSECS)*, vol. 2, no. 1, pp. 9–17, May 2022, doi: 10.35870/ijsecs.v2i1.761.
- [3] A. Shah, H. Vasudavan, and N. F. Razali, "Event management systems (EMS)." A. S. Sahu, "Online event management system."
- [4] M. V. Bramhe, M. A. Waghmare, B. Awate, K. Rao, A. Kadu, and T. Dangre, "Online event management system: A critical review of research findings and methodologies," vol. 9, no. 5, 2024.
- [5] N. S. M. Yusoff, N. A. A. Bakar, and N. M. N. Daud, "The development of a preschool event management system using usability heuristic theory," *IOP Conference Series: Materials Science and Engineering*, vol. 1176, no. 1, p. 012038, Aug. 2021, doi: 10.1088/1757-899X/1176/1/012038.
- [6] M. K. Kabier, A. A. Yassin, Z. A. Abduljabbar, and S. Lu, "Role-based access control using biometric in educational system," *Basrah Journal for Engineering Sciences*, vol. 49, no. 1, pp. 85–101, Jun. 2023, doi: 10.56714/bjrs.49.1.8.
- [7] M. A. U. Marzele and R. A. Jalil, "Web-based UTHM event management system."
- [8] R. O. Obe and L. S. Hsu, *PostgreSQL: Up and Running: A Practical Guide to the Advanced Open-Source Database*, 1st ed. Beijing, China: O'Reilly, 2012.