WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 4, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v4n3.47
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Competition Registration and Announcement Platform Based on Node.js and .NET Core Web API Development

Yuanyuan Zhang *

School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, China

ABSTRACT

In this project, the front-end is developed using the Vue template engine based on Node.js and the Element UI component library, while the back-end is built with .NET Core Web API. SQL Server is used as the database. The system's front-end is primarily designed for use by regular users, including both teachers and students, who can log in only after applying for registration and passing approval. Regular users have access to various modules, such as the event release module, research project module, news information module, notice module, and personal center module. On the back-end, administrators have access to several management functions, including competition module management, team module management, research module management, system module management, user module management, and achievement module management. The event registration and announcement platform streamline communication and reduces costs for teachers and students in the team formation process by centralizing the management of events and projects, which are typically difficult to oversee. Additionally, it allows users to independently maintain event and project information, reducing the workload for platform operators.

KEYWORDS

Competition Registration and release platform; Team formation; Net core web API; SQL Server

1. INTRODUCTION

With the rapid development of society and the general improvement in cultural and educational levels, contemporary university students no longer limit themselves to learning solely through classroom instruction [1, 2, 3]. In their spare time, they actively seek opportunities for growth—not just by participating in entertainment activities, but also by aiming to enhance their skills, broaden their experiences, and expand their social networks. These efforts serve to better prepare them for their transition into the professional world [4, 5]. Increasingly, students choose to participate in various competitions or research projects as a means of achieving these goals. Before engaging in a competition or research project, a critical step involves assembling a team. This typically begins with understanding the details of the competition or project and disseminating this information widely. The next step is finding like-minded peers to collaborate on the endeavor. This process requires extensive communication and coordination, as well as effective outreach to potential team members. Such efforts often extend beyond the confines of a single class, academic year, or even department [6, 7, 8, 9].

In today's information age, traditional methods of communication—such as class meetings, small-scale group chats on platforms like QQ or WeChat—are no longer efficient or targeted enough to meet these needs [10, 11, 12, 13]. To address these challenges, there is a pressing need to develop a dedicated online platform. This platform would serve as a specialized hub for accessing information

about competitions or research projects, registering for participation, and forming teams. By doing so, it could significantly reduce the communication overhead for both students and teachers, enabling them to achieve their goals more efficiently [14, 15].

2. PLATFORM ANALYSIS

The competition registration and announcement platform consist of two main components: the front-end user interface and the back-end management system. Within the front-end interface, users are categorized into two roles: visitors and regular users. Visitors, who are unregistered users, can only browse publicly available information on the website. They have the option to register as users, at which point they become regular users. Regular users, upon passing an administrator's review and verification process, gain access to the platform's full functionality by logging in.

The front-end interface offers users several key features, including a competition announcement module, a project posting module, a news and updates module, a notifications and announcements module, and a personal user center module. On the other hand, the back-end management system is designed to provide administrative capabilities such as competition management, project management, notifications and announcements management, user management, team management, and achievement management.

Before logging in, regular users are considered visitors, during which they can browse competition announcements, project information, news updates, and notifications. They may choose to register as users, but login access is granted only after receiving approval from the back-end administrators. Once logged in, users gain access to manage several functional modules, including the competition announcement module, research project module, news updates module, notifications module, and personal center module.

Taking the management of the personal center as an example, its use case can be analyzed as follows: Users can navigate to the personal center to update their personal information. Additionally, they have the option to view or delete outcome displays from competitions or research projects they have participated in or collaborated on. This feature enhances user autonomy in managing their profiles and project achievements effectively.

When a back-end administrator logs into the management system, they are granted full control over several key modules, including the competition management module, project management module, notifications module, user management module, team management module, and achievement management module. Administrators have comprehensive permissions for these modules, including the ability to add, delete, edit, and manage all related information.

For instance, in the case of project information management, the use case can be described as follows: After logging into the back-end management interface, the administrator can navigate to the project information management tab. From there, they can perform a range of operations, such as querying, adding, editing, and deleting project information. This functionality ensures efficient and centralized management of all project-related data, enhancing the overall administration process.

3. PLATFORM DESIGN

3.1. Platform Database Design

Due to business requirements, the system is designed with nine entity classes, with each table in the database corresponding to one entity class. After careful analysis and consideration, the relationships between these entity classes are illustrated in the Entity Relationship Diagram (ERD) shown in Figure 1.

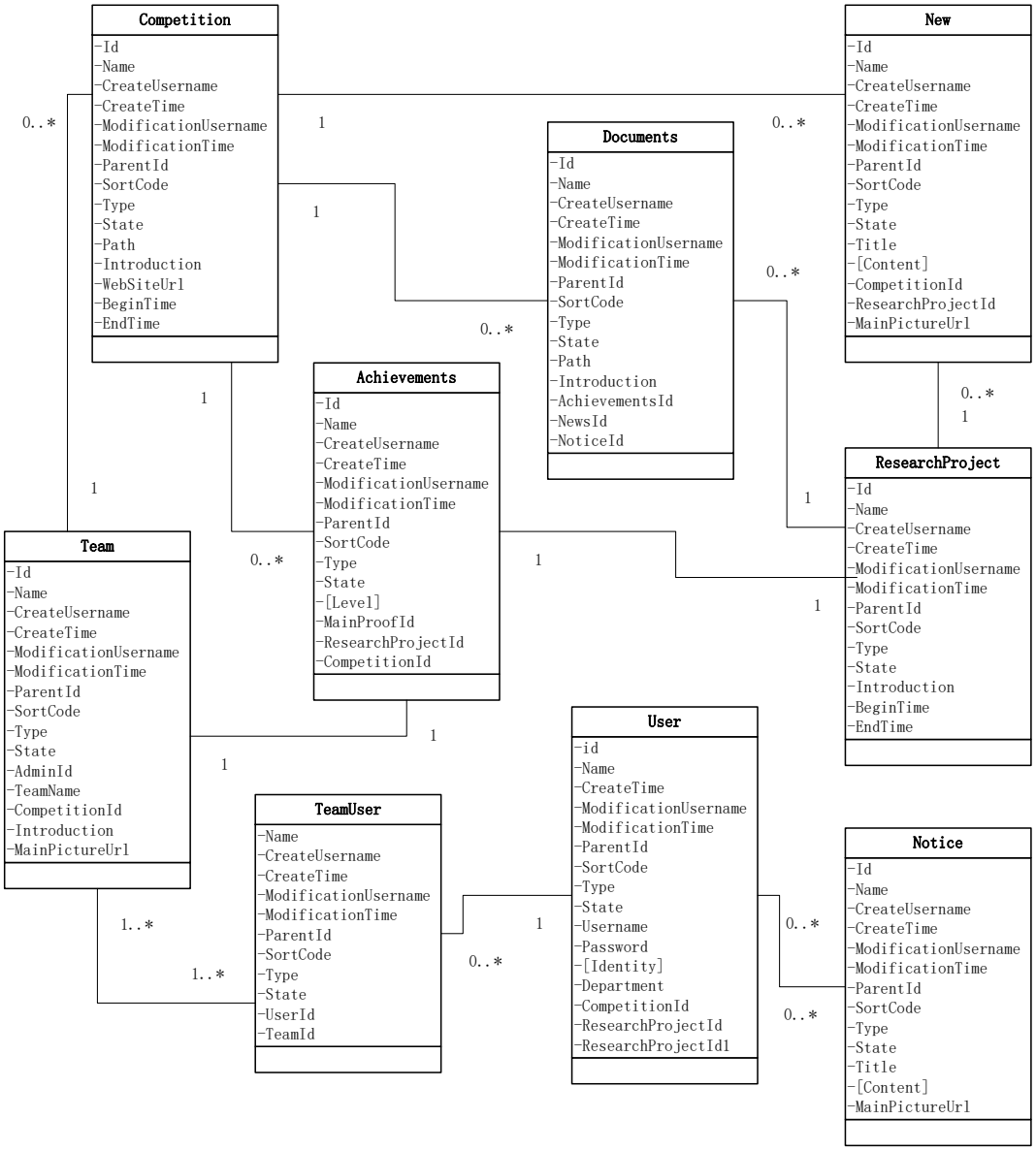

Figure 1. Class Diagram of Entity Classes

3.2. Competition Announcement Module

The Competition Announcement Module is one of the core modules of the system. Its primary function is to display a list of existing competition sections, allowing users to browse through available events. If users cannot find the competition they are looking for, they have the option to create a new competition section themselves. Furthermore, the creator of a competition section is able to maintain and update the information related to their event as needed.

In this section, we will provide a detailed design and analysis of the functionality that allows users to create their own competition sections. This feature is essential for providing flexibility and ensuring that the platform can accommodate a diverse range of events. By enabling users to create and manage competition announcements independently, the system empowers participants and organizers, fostering a dynamic and user-driven environment.

First, when a user enters the competition announcement module of the system and, after filtering, cannot find their desired competition section, they can click the "Create Competition" button located at the bottom-right corner of the homepage of the competition announcement module. This action will invoke the AddCompetition method. The method first checks whether the user is logged in. If

the user is already logged in, the system will use the CustomizeRoute method to directly redirect them to the competition section creation page. However, if the system detects that the user is still in a visitor state (i.e., not logged in), it will prompt the user to log in before proceeding.

3.3. Project Announcement Module

The Project Announcement Module is also one of the two core modules of the system. While it shares similar features with the competition announcement module—such as the ability to create sections and display outcomes—the logic behind team formation in this module differs fundamentally from that of the competition module. In the competition module, each competition section can host multiple teams, and it is necessary to manage the status of these teams. In contrast, in the project module, each project can only have one team, and the initiator of the project serves as the team leader.

In this section, we will focus on the design and analysis of the team formation functionality within the project announcement module. This feature plays a crucial role in ensuring that project teams are formed efficiently, with the project leader overseeing and managing the team. Understanding the differences between the competition and project modules, particularly in terms of team structure, is essential for effectively utilizing both modules.

When a user finds a project they are interested in and wishes to join a team, they can use the "Join Team" functionality in the Project Announcement Module. The user first enters the project section homepage. When the user clicks the "More" button at the bottom-right corner of the project section interface, the el-DropList component will be displayed. From the popup menu, the user can select the "Apply to Join" option to send a team membership request to the project leader.

After the user clicks the "Apply to Join" button, the addTeam method will check if the user is logged in. If the user is not logged in, the system will prompt them to log in before proceeding. If the user is already logged in, the front-end will retrieve the user's identity information (such as accessList) stored in the browser from the login session. The system will then send the user's identity information (userId) to the back-end server to request joining the team.

Upon receiving the request, the server-side TeamUpController will insert the user as a pending member into the team list. It will then return a success message to the front-end, which will be displayed on the interface. The project creator will be able to see this new member in the team list. When the project creator clicks "Agree to Join" (agreeTeamUp), the member will officially join the team. If the project creator clicks "Deny" (denyTeamUp), the member will be automatically removed from the team list.

3.4. News Management Module

The News Management Module is primarily designed to manage news content published in competition sections and event areas created by front-end users. If any article content is deemed inappropriate or problematic, the system administrator has the ability to promptly delete the news article.

When the system administrator identifies an issue with a news article, they can click the "Delete" button in the operations section of the news list. This button triggers the click event bound to the el-button component in the front-end, which calls the handleDelete function. The function retrieves the ID of the selected article and sends it to the back-end server. Upon receiving the delete request, the server-side NewsController will call the delete method from the NewsService. The NewsRepository will then interact with the database to delete the article. Finally, the operation result will be returned to the front-end and displayed to the administrator.

3.5. Team Management Module

This module is designed to provide system administrators with an interface to query the team formation status in various competitions. Within this interface, administrators can view the teams formed in each competition. If any teams are found to violate the rules, administrators have the option to delete the teams.

When an administrator identifies an abnormal team, they can click the "Delete" button in the operations area on the right side of the team list. Upon clicking, the click event is triggered, and the delete method is invoked. This method retrieves the ID of the team to be deleted and calls the delete method in the TeamUpController on the server-side. The controller then calls the corresponding Delete method in the TeamUpService, which performs the deletion operation on the database.

If the team has uploaded corresponding results, the team cannot be deleted, and a failure message will be returned. However, if the team has no associated results, it will be deleted successfully.

4. SUMMARY

This system is built using the .NET Core Web API framework, with a focus on using Web APIs for development, thereby achieving a separation of the front-end and back-end. The front-end is responsible for page styling, dynamic data parsing, and rendering, while the back-end focuses on implementing business logic. Data interaction between the front-end and back-end is accomplished through APIs.

The back-end adopts a layered design, clearly categorizing the back-end components. Each library has its own independent role and is isolated from others, with interactions between layers taking place through API calls to exchange data. On the front-end, the system utilizes the Vue.js template engine based on Node.js and the Element UI style library. Vue.js is one of the most popular front-end development frameworks today, while the Element UI component library offers visually appealing, feature-rich components that are easy to use and simple to operate. This allows for a smooth and efficient front-end development experience, enabling developers to focus more on system logic. All the front-end technologies used are open-source, which reduces costs and increases development efficiency.

REFERENCES

[1] Muradov O. Basic principles and rules of innovative pedagogical technologies in the educational process [J]. M·odels and methods in modern science, 2024, 3(1): 84-93.

[2] Green M D. A new web application for determining sample size in freedom-from-disease testing with imperfect tests [J]. Preventive Veterinary Medicine, 2025, 235106397-106397.

[3] Chen X. Exploring the Learning Engagement of Digital Media Art and Digital Media Technology Students: A Case Study of the Web Design and Development Course at Nanfang College, Guangzhou [J]. Journal of Educational Research and Policies, 2023, 5(12).

[4] Revolutionizing Digital Presence: Elevating Businesses with Premier Website Design Services Conway AR [J].M2 Presswire, 2023.

[5] Eszter C, GuyBart S. Parsley: a web app for parsing data from plate readers. [J]. Bioinformatics (Oxford, England), 2023.

[6] Xinghui H, Yanlei M, Wuhao Z, et al. Radial envelope forming mechanism and process design method for cylindrical rings with thin wall and high web ribs [J]. Chinese Journal of Aeronautics, 2023, 36(12):461-476.

[7] Xuan H H, Tuan A T, Phuc N N, et al. Monitoring and control system of environmental parameters in swiftlet houses [J]. Internet of Things, 2023, 24.

[8] Jie W, Xin H, Tianjun J, et al. A Web-based coordinated control platform for source-grid-load-storage of low-voltage stations [J]. Journal of Physics: Conference Series, 2023, 2666(1).

[9] E. S W, Lucy W, Rhys A, et al. Co-Designing an Air Quality Web App with School Pupils and Staff: The SAMHE Web App [J]. Citizen Science: Theory and Practice, 2023, 8(1):64-.

[10] Clio Websites Emerges as the Foremost Authority in Custom Web Design and Development, Transforming Business Digital Presence in Calgary [J]. M2 Presswire, 2023.

[11] Sandnes E F. To wrap or not to wrap? A study of how long words are split when reflowed on magnified web pages [J]. Universal Access in the Information Society, 2023, (prepublish):1-13.

[12] N A J, Limin W, Susan B, et al. Quality By Design Pilot Analysis of FDA Regulatory Guidance Web Data Usability for Innovators in Non-Hodgkin Lymphoma: Overcoming Challenges to CAR-T Development [J]. Blood, 2023, 142(S1):7169-7169.

[13] Quentin L. Continuity and discontinuity in web archives: a multi-level reconstruction of the; firsttuesday; community through persistences, continuity spaces and web cernes [J]. Internet Histories, 2023, 7(4):354-385.

[14] Gatheeshgar P, Bock M, Chandrasiri D, et al. Assessment of Eurocode 3 Shear Design Provisions for Cold-Formed Steel Beams with Web Holes [J]. ce/papers, 2023, 6(3-4):1983-1988.

[15] Saule K D, Lyubov N K, Nurlan A M, et al. Implementation of a Web Application and GIS Electronic Atlas for Teaching in Open Education [J]. International Journal of Information Technology and Web Engineering (IJITWE), 2023, 18(1):1-23.