**《Multi-campus Student Management System Report》**

**Author：Junnan Zhao**

**Id：23010566**

**My GitHub URL:**

**Code files, reports, and videos are all available in the GitHub repository, please click on the link above to access them.**

**My video URL:**

**My video was posted on BiliBili.**

**I. Project structure文本

描述已自动生成**

**1. Overall layout of the project**

**o the project root contains multiple folders, such as "Controllers," "Models," "Scripts," and "Views," which contain code and files for different feature modules.**

**o the "Connected Services" and "Properties" folders may be used for project configuration and files related to connecting to external services.**

**2. Controllers folder**

**o This folder contains several controller class files, such as "CampusController.cs", "HomeController.cs", and "StudentsController.cs".**

**o These controller classes are responsible for handling HTTP requests from the user, calling the appropriate business logic, and returning views or data to the frontend. For example, "StudentsController.cs" handles requests for actions related to students such as getting student list, adding student, editing student information, etc.**

**Models’ folder**

**o Contains two model class files "Campus.cs" and "Student.cs".**

**The o model class is used to define data structures that represent the table structure in the database. For example, the "Student.cs" class contains the attributes of the student such as name, campus, class, etc., which correspond to the fields of the "Student table" in the database.**

**Scripts folder**

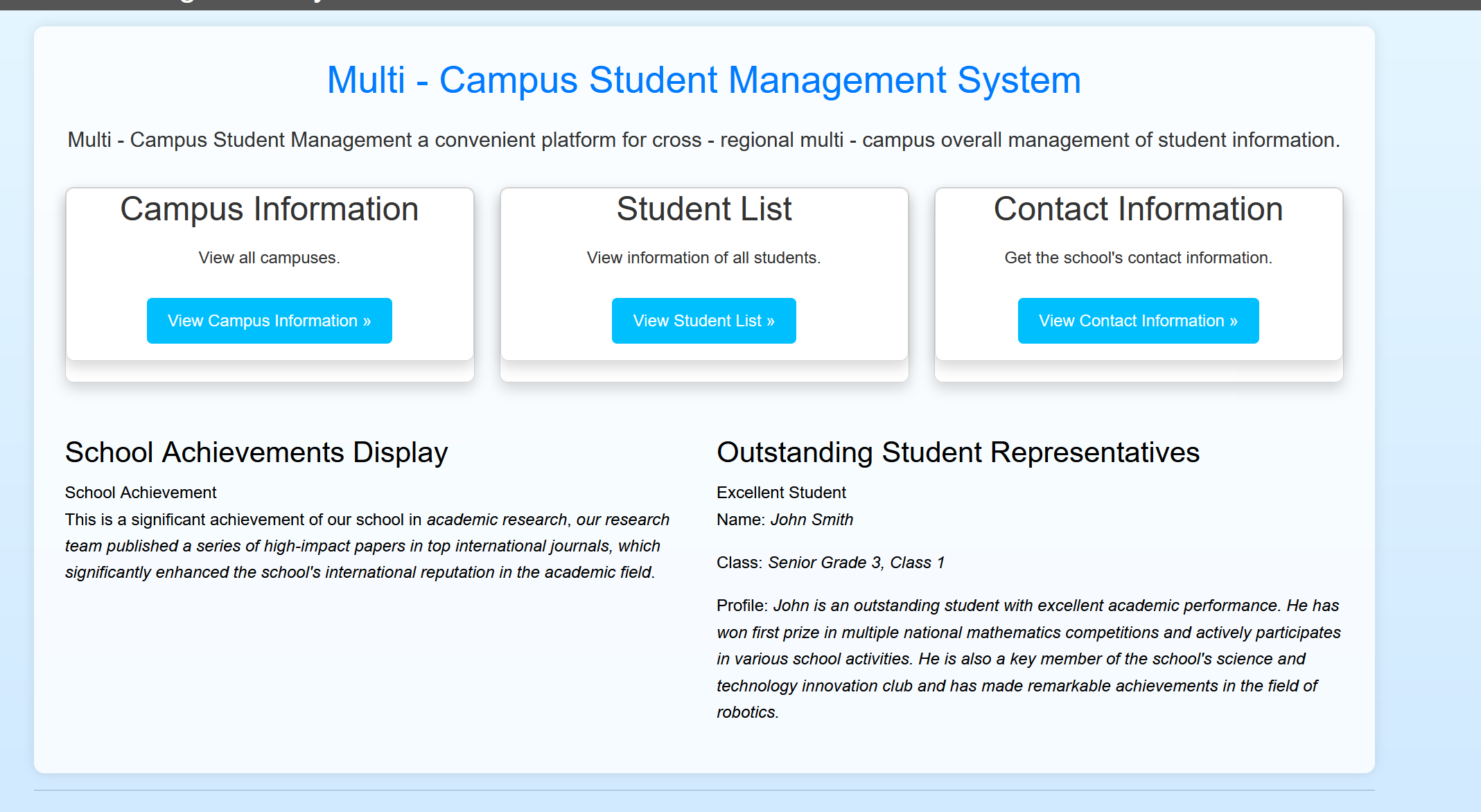
**o This folder is usually used for JavaScript files that are used to implement the interactive logic of the front-end. Scripts for form validation, asynchronous data fetching, and more.**

**5. Views folder**

**o There are multiple subfolders under the views folder, such as "Campus" and "Students", corresponding to the view files of different modules.**

**o Each subfolder contains multiple. cshtml files, such as "Create.cshtml", "Delete.cshtml", "Details.cshtml", "Edit.cshtml", and "Index.cshtml". These view files are used to render the user interface, such as "Index.cshtml" in the "Students" folder for the student list page and "Create.cshtml" for the form page to add student information.**

**ii. System function**

**Main page**

It realizes the introduction of the basic functions of the website and supports the direct jump to the website functions. In addition, it has developed the school achievement display and excellent student display section, which enriches the content of the website.

Contact Page图形用户界面, 文本, 应用程序

描述已自动生成

**Student listing page（Index.cshtml）**

图形用户界面, 应用程序

描述已自动生成

**o Page layout**

**o There is a navigation bar at the top of the page that shows the current page as "Students" and a "Create New" button to add new student information.**

**o There is a data table in the middle of the page, and the table column names include Campus Name, Student Name, and Address.**

**o Each student record line also contains "Edit", "Details" and "Delete" operation links to facilitate user management operations on student information.**

**o Data filtering and sorting**

**o page provides data filtering and sorting functions. Users can select a specific campus to Filter data through the "Filter by campus" drop-down menu or select the sorting method through the "Sort By" drop-down menu.**

**o the page currently shows "Showing all 4 students", which indicates that the system is able to obtain and display the student data correctly.**

**o and the system can implement paging operations**

**2. Action functions (Create, Edit, Delete, Details)**

**o Add students (Create)**

**o by clicking the "Create New" button, the user can go to the "create. cshtml" page to add new student information. In this page, the user can input the information of the student, such as the campus, name, class and so on, and then submit the form, and the system will save the new student information to the database.**

**o Edit student information (Edit)**

**o After clicking the "Edit" link, the user will be taken to the "Edit.cshtml" page, which will display the existing information of the current student and allow the user to modify this information. After the modification is completed, the form will be submitted, and the system will update the student record in the database.**

**o View student Details**

**o the "details. cshtml" page is used to display the Details of a student, and the user can click on the "Details" link to view the full information of a particular student, which may include other additional attributes of the student, such as contact information.**

**o Delete the student (Delete)**

**o After clicking the "Delete" link, the system will pop up a confirmation dialog box, and the records of the corresponding students will be deleted from the database after confirmation.**图形用户界面, 文本, 应用程序, 网站

描述已自动生成

The operation functions of the Campus interface are as above. In addition, the correlation and independence between campus and students are technically realized. When a whole campus is deleted by mistake, the internal student data will be retained.

Iii. Technical implementation

Backend technologies (ASP.NET)

o Use ASP.NET framework to build the backend logic文本

描述已自动生成

**1. Front-end technologies (HTML, CSS, JavaScript)**

**o the front-end page uses HTML to build the page structure, CSS for style design, custom CSS template to achieve the design of the website, respectively design the navigation bar and page**文本

描述已自动生成

文本

描述已自动生成

o JavaScript is used to implement interactive functions such as filtering, sorting, and form submission. See GitHub for the code文本

描述已自动生成

* + **Iv. System advantages**
  + **1. Functional integrity**
  + **o system has the basic functions of student information management, including adding, editing, deleting and viewing details, to meet the daily needs of multi-campus student management.**图形用户界面, 文本, 应用程序

    描述已自动生成
  + **2. Ease of data manipulation**
  + **o provides data filtering and sorting, pagination function, convenient for users to quickly find and manage the specific campus or a specific order of student information, improve the management efficiency.**图形用户界面, 文本, 应用程序

    描述已自动生成

**V. Summary**

**The multi-campus student management system has certain rationality and practicability in the project structure and function realization. The back-end logic is realized by ASP.NET framework, and the front-end interface and interaction are realized by combining HTML, CSS and JavaScript, which can meet the basic needs of student information management. However, there is still room for improvement in terms of user experience, performance and functional extension, and the system can be made more perfect and efficient through further optimization and extension.**