Design and Implementation of Online Education Platform Based on React

Wang Chen[a], Zuxu Dai[b]*

{[a]984331807@qq.com, [b]zxdai@wit.edu.cn}

School of Mathematic and Physics, Wuhan Institute of Technology ,WUHAN ,China

Abstract. With the development of the Internet, online education has become a trend, especially during the epidemic. Therefore, it is particularly important to develop an online education platform. An online education platform was developed based on the React framework, which took advantage of the componentization of the React framework, improved development efficiency, improved user experience and performance, and realized page componentization. The shortcomings of native JS code redundancy are optimized, and problems such as difficult project management and poor scalability are solved.

Keywords: Online Education Platform, React, Componentization.

1 Introduction

Online education has become one of the hottest fields in the current digital age. In the new crown pneumonia epidemic, online education platforms have played an important role in the teaching work of students [1]. With the support of modern technology, more and more educational institutions and individuals choose to conduct online education business on the Internet [2].

In this context, this paper proposes a React-based design and implementation of an online education platform. As one of the most popular front-end development frameworks [3], React is efficient, concise, and flexible, and can well meet the needs of online education platform development [4]. This paper first introduces the requirement analysis and functional design of the online education platform, and then elaborates on the front-end architecture design [5] and database design [6], as well as the solutions to problems encountered during development.

The purpose of this article is to provide an efficient and reliable technical solution for online education platform developers to help them quickly develop an online education platform that meets user needs.

2 Project overview

Both the front and back ends of the project are developed on Vscode [7]. The front end uses the current mainstream React framework, the back end uses Golang, and the database uses MySQL. The project adopts the architecture model of front-end and back-end separation [8],

EIMT 2024, March 29-31, Wuhan, People's Republic of China

DOI 10.4108/eai.29-3-2024.2347638

and separates the development, deployment and maintenance of the front-end and back-end codes, so that the two can evolve and upgrade independently, improving development efficiency. The front and back ends can be expanded and modified separately, which improves the scalability. To improve the code reuse rate, the front end uses React-Router to realize page jumping [9], and calls the back-end service through the API interface, reducing repeated code development.

3 Demand analysis

3.1 Functional requirements

The system includes student terminal, teacher terminal and management terminal.

The student terminal includes the following functional modules

- In the login registration module, registered users can log in directly. Unregistered users will be prompted to be unregistered and jump to the registration page.
- The shopping cart module supports settlement, viewing, and deletion of courses that have been added to the shopping cart, and supports batch operations such as selecting all/unselecting, batch deleting, and one-click settlement.
- The learning center module includes functions such as viewing personal information, my order, my class schedule, and learning statistics.

The teacher's function includes the following functional modules:

- Login registration function, you can log in with account number and password, mobile phone number and verification code, and WeChat one-click login.
- Course management module, features include

Course release: Teachers can create courses online, fill in course information, courses, and courseware prices, and then upload the courses to the background for review. After passing the review, they will be displayed on the student side; The class time is scheduled, and students can choose the class time that suits them according to the teacher's course scheduling time

Course management: support addition, deletion, modification and query of course information; courses that have already taken effect cannot be modified or deleted; courses that have not taken effect can support information modification and deletion; and live courses are supported. Teachers can initiate course live broadcast online, and live room information can be Sync to student schedule.

- Personal center module, you can check your personal information, course statistics, remuneration calculation, etc.

The management terminal includes the following functional modules:

- The user management module can manage the list display of users, and can view and filter user information. Check and manage the basic information of students and teachers, and the association of order information.

- The live broadcast management module can perform live broadcast authority distribution (whether the user has permission to perform live broadcast), live broadcast data management, manage live broadcast playback, video data, etc., whether it supports downloading to the local, and view live data.
- The data report module can manage and view data reports and sales data, and has a clear statistics on platform data.
- Feedback management module, you can view user feedback information.

3.2 Non-Functional requirements

- The reliability of the system. The online education system is composed of front-end, back-end, database, network, etc. The front-end adopts the React framework to provide the user interface, the back-end uses Golang to implement the core logic, the database uses Mysql to store data, and the network connection uses It is used to connect the front end and the user. Therefore, the reliability of the overall system depends on the reliability of these components.
- System stability, the online education system needs to ensure long-term stable operation, which is also an important indicator of system reliability. The stability of the system includes the system's fault tolerance, load capacity, and concurrency capabilities. When the system fails or crashes, an effective fault tolerance mechanism is required to ensure that the system will not crash due to a single point of failure.

4 Project design ideas and implementation

4.1 Project design ideas and implementation

The system adopts the development mode of front-end and back-end separation. The front-end uses the React framework, and the back-end uses Golang. In order to meet the required functions of online education, three modules are designed: the student end, the teacher end, and the management end, as shown in Figure 1.

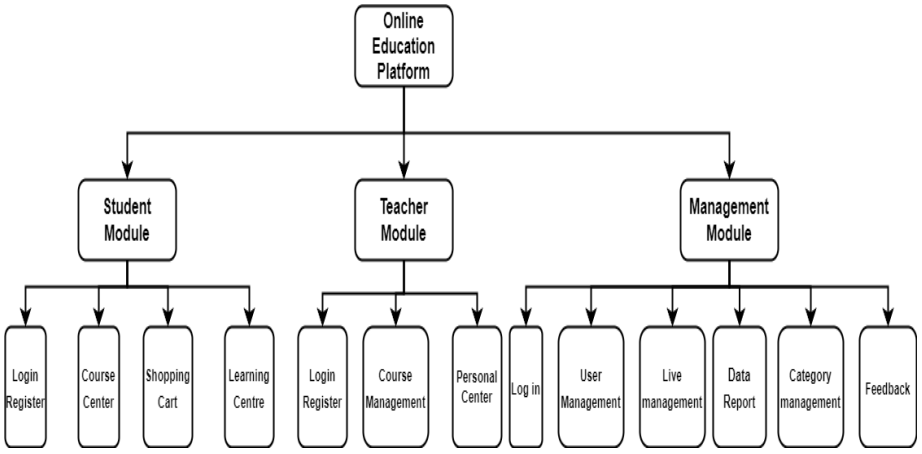

Figure 1. Functional module diagram

4.2 Front-end architecture design

The overall architecture of the online education system is shown in Figure 2. Based on the React componentization framework, it is divided into business components, basic components, dependency layers, and scaffolding layers from top to bottom. The business component is user-oriented, and the layer that the user has intuitive contact with. According to the role of the registrant, it is judged whether the information of the registrant is a teacher, a student or a manager, and then the corresponding business function is granted.

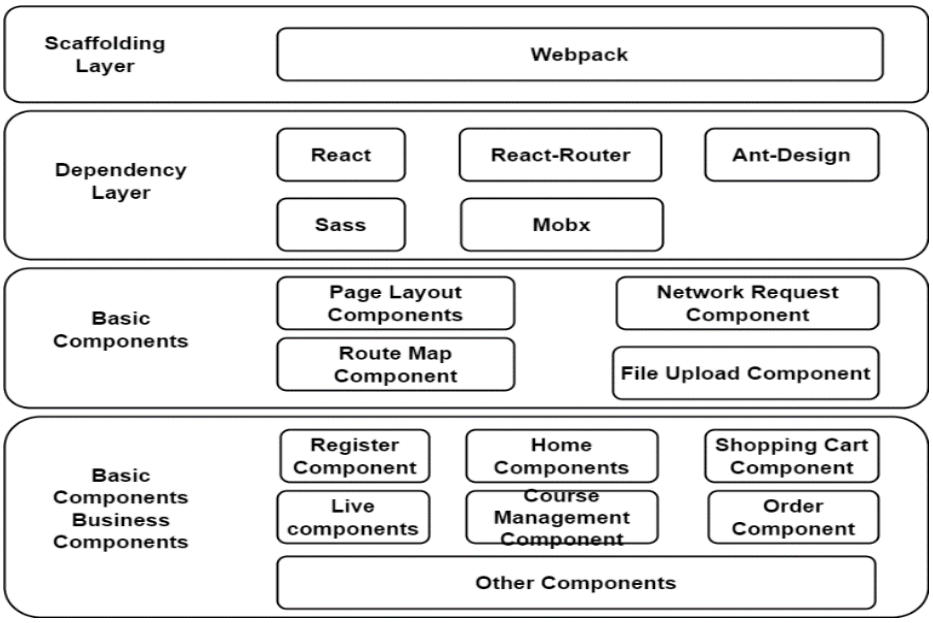

Figure 2. Front end architecture diagram

Business components include registration components, shopping cart components, live broadcast components and order components. Different components are responsible for the corresponding functional logic, and connecting all the components in series according to the business process is the online education platform.

Basic components mainly include page layout components, network request components, route mapping components and file upload components. Among them, the page layout component is a component that dynamically generates pages according to business characteristics, and the other components provide services such as data requests and page jumps for upper-level business components.

The dependency layer is the information in the package.json file in the root directory of the project. It customizes the required modules and project configuration information (such as name, version and other metadata). According to this configuration file, through the npm install command, it will automatically Download the required modules. The dependency layer is the skeleton of the project, and the development of basic components and business components based on it cannot be separated from the support of the dependency layer.

The scaffolding layer mainly uses Webpack to provide packaging and building functions for the client, which can convert, package and compress existing class resources into static resources. In addition, it also provides functions such as local server construction and hot loading, which provide strong support for developers to optimize development and optimize output quality.

5 Technical points under development

5.1 Eliminate asynchronous contagion

JavaScript divides task execution modes into synchronous and asynchronous. Asynchronous operations are often involved in the transfer of data between components. If not handled properly, it may lead to asynchronous contagion, which in turn affects the performance and stability of the entire application. The asynchronous functions async and awit in ES7, if you use await, you must use async. If there is no cache, an exception Promise will be thrown, which will block the following code. This article adopts React's component development idea, and uses Suspense components to eliminate asynchronous contagious. Components are shown in Figure 3.

```
<Suspense fallback={"loading"}>
  <Lazy/>
</Suspense>
```

Figure 3. <Supense> Component diagram

5.2 Routing

This paper develops React-Router, a routing library based on React, to realize page jumping. Using React Router, first define multiple routing components, which will be mapped to different URLs and render the corresponding UI at the same time. Create a Router component in the component, and pass a routing configuration object, which contains the URL to be mapped and the corresponding routing component. When the user enters a URL in the browser or clicks a link on the page, React-Router will match the corresponding routing component according to the rules defined in the routing configuration object. After matching the corresponding routing component, React Router will render the component to the page and display it to the user. When the user jumps to another route in the page, React Router will update the URL according to the rules defined in the route configuration object. The specific code of the routing jump in this article is shown in Figure 4 below:

```
function App() {
  return (
    <div className={classNames('app', 'override-antd-tongtong')} id={'override-antd-tongtong'}>
      <BrowserRouter basename='/iot-platform'>
        <Switch>
          <Route exact path='/login' component={Login}></Route>
          <Route path='/liveRoom' component={LiveRoom}></Route>
          <Route path='/' component={Index}></Route>
        </Switch>
      </BrowserRouter>
    </div>
  );
}
```

Figure 4. Routing Code Diagram

5.3 Cross-domain issues

In order to ensure the smooth calling of the API between the client and the server, this article encapsulates the GET, PUT, POST, DELETE, and PATCH methods, but if there is any one of the domain name, protocol or port between the client and the server At the same time, a cross-domain problem occurs. Although the cross-domain request can be initiated normally, the returned result is intercepted by the browser. In order to solve this problem, this article adopts the CORS method at the back end, and only needs to call normally at the front end. The AXIOS method will do.

6 System implementation

After the project development was completed, the website was tested using the Chrome browser and there was no abnormality, and the page display and user interaction were in line with the requirements.

7 Conclusion

This paper discusses the design and implementation of an online education system based on the React framework, and mainly introduces project demand analysis, database design, and technical implementation. The performance is improved through the use of front-end and back-end separation, componentization, and elimination of asynchronous infection.

The advantages of the React framework have been verified through practice. In addition, after testing, although the webpage designed in this paper has achieved the expected goal, it needs to be improved in terms of security. The next step is to use the Shiro security mechanism to authenticate user permissions to improve security.

Acknowledgments. National Natural Science Foundation of China (No.61771353)

References

[1] Fu Weidong, Zhou Hongyu Challenges and coping strategies brought by the COVID-19 to China's online education [J] Journal of Hebei Normal University2020, 22(2): 14-18.

[2] Tu Bo, Gong Wenli. New features of online education development in the context of mobile internet [J]. Modern vocational education,2020(18):16-17.

[3] Bhalla A, Garg S, Singh P. Modern web development with reactjs [J]. International Journal of Engineering and Technology Research, 2020, 7(05):18-21

[4] Dong Ming. Research on Main Problems and Countermeasures in Computer Software System Development[J].Network Security Technology and Application,2021(11):51-52.

[5] Ni Chuang. (2020). Research on the Design Method of Front-End Architecture of Video Websites (Master'sThesis,BeijingJiaotongUniversity).https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD202101&filename=1020444748.nh

[6] DuBois P. MySQL[M]. Pearson Education, 2008.

[7] Code V S. Visual studio code[J]. Recuperado el Octubre de, 2019.

[8] Wang Jian, Luo Zheng, Zhang Xi, et al. The design and implementation of front-end and back-end separation of Web projects[J]. Software Engineering, 2020, 23(4): 22-24.

[9] Nguyen H. Single-page application and frontend testing methods: built with React and React Router, tested with Jest and Cypress[J]. 2022,52(18),15-17