International Journal of Engineering Education Vol. 40, No. 2, pp. 231–239, 2024
Printed in Great Britain
0949-149X/91 $3.00+0.00

Design and Implementation of a Wechat Mini Program for Remote Teaching of Control System Experiments*

KEZHI LI[1], GUO-PING LIU[2] and WENSHAN HU[3]
[1] School of Electrical Engineering and Automation, Wuhan University, Wuhan, China. E-mail: likezhi1999@foxmail.com
[2] Center for Control Science and Technology, South University of Science and Technology, Shenzhen, China. E-mail: liugp@sustech.edu.cn
[3] School of Electrical Engineering and Automation, Wuhan University, Wuhan, China. E-mail: wenshan.hu@whu.edu.cn

With the increasing number of mobile phone users, this paper proposes a mobile experiment platform software with a hybrid frontend architecture of web and Wechat mini program. After the software being developed and deployed, a group of students participated in the experiment of controller design over a DC motor angle control system online on the Wechat mini program. A survey was carried out to gather opinions on the user experience and the educational value for the course "Principles of Automatic Control". According to the results of the survey, about 80% of the students completed the design of the controller and thought the result was satisfactory. The survey results also shows that over 90% of the students thought it helped their understanding of the course and that they gained controller design experience.

Keywords: mobile learning; remote laboratory; Wechat mini program; networked control

1. Introduction

Since the 1990s, both traditional distance education institutions and conventional universities have started to utilize the Internet, marking the beginning of an era e-learning or the third generation of distance learning[1]. Mobile learning is an extension of e-learning in the era of mobile internet, enabling learners to engage in interactive learning anytime and anywhere through mobile devices and technologies [2]. In the era of e-learning, with the development of internet technologies, web-based remote laboratories have been continuously developed and applied in university teaching activities [3–6]. Unlike other remote laboratories, the Networked Control System Laboratory (NCSLab) is a web-based remote laboratory that provides control system experiments for undergraduate students [7]. But recently, people are spending more time on mobile phones and mobile applications. Therefore, this paper introduces a remote experiment operating software based on mobile platforms for NCSLab.

Compared to other types of mobile learning applications [8–11], the NCSLab Wechat mini program provides remote laboratory education on control theory for university students. With mobile learning technology, the NCSLab mobile platform can serve thousands of students without significantly increasing resource inputs to the system. Furthermore, the application of mobile technology in education detaches learning activities from traditional teaching settings which means that the NCSLab mobile platform can provide college students with access to experiment resources anytime and anywhere.

The current technological approaches for developing mobile learning software are web application, native application or hybrid application [8-11]. Based on the deployed technologies of NCSLab PC platform, this paper proposes a design and implementation of hybrid software as Wechat mini program. Compared to native software or other hybrid architecture software [12], the WeChat mini-program saves developing time, supports multi-platform including iOS and Android, and requires no download.

2. Analysis of NCSLab Mini Program

2.1 Structure Analysis of NCSLab Server

The NCSLab server has already been built as developing the NCSLab website [13]. It provides with remote laboratory services for automation engineering and control theory experiments. As shown in Fig. 1, a server cluster consisting of Nginx server, static resource server, PHP server, experiment server and camera server collectively provides services for the NCSLab mini program as well as the website.

The Nginx server does the job of reverse proxy and load balance. It receives HTTP requests or WebSocket requests from the app and then dispatches the request to one of the servers according to the request's URL and the routing rules [14]. Requests for static resources such as HTML, JavaScript, CSS files, images, videos and so on goes to the static resource server. Requests for informations of the user account, the experiment plant, or the control algorithms goes to the PHP server. After receiving the reuqest, the PHP server carry out data manipulations on the MySQL database and returns the results to the user [15, 16].

* Accepted 2 December 2023.

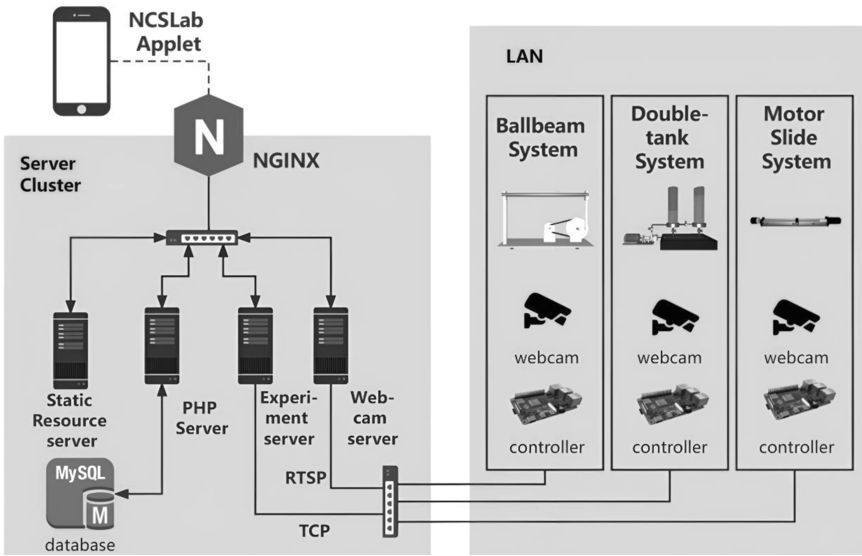

Fig. 1. Architecture of the Server Cluster.

Before starting the experiment, NCSLab mini program must establish WebSocket connection to the experiment server. As shown in Fig. 1, there are TCP connenctions between the experiment server and the experiment plants. During the experiment, the experiment server collects and caches the experiment data from the experiment plants and then sends them to the NCSLab mini program. At the same time, the webcam server also collects video streams from the webcam in the experiment field and transmit them to the NCSLab mini program.

NCSLab provides users with a wide range of control plants for experiment, including ballbeam system, dual-tank system, motor slide system, fan speed control system and so on. The networked controller collects and processes real-time experiment data generated by the sensors including the PWM speed wave of the fan, speed data from serial communications with absolute encoder, electric signals from ultrasonic distance tester, and voltage signals from weight sensors. Furthermore, the networked controller calculates and generates PWM control signals to drive the movement of the experiment plant.

Based on the existing technologies and hardware foundation mentioned earlier, a mobile-friendly NCSLab mini-program can be developed through a redesign of the frontend. Compared to the NCSLab website and NCSLab mobile client, this mini-program will provide remote laboratory services to mobile users without requiring them to download a separate app. Moreover, unlike the NCSLab mobile client, the mini-program will be accessible to both iOS and Android users, providing the same remote laboratory services. Technically, the development of the mini-program will primarily focus on the design of the mini-program framework and user interface (UI). The goal is to achieve the app's deployment with minimal development costs while offering practical remote laboratory services to users. By utilizing the mini-program framework and optimizing the UI for mobile devices, the development process can be streamlined, and users can conveniently access the remote laboratory services through their mobile browsers without the need for app installation.

2.2 Function Analysis of Wechat Mini Program

Although the server cluster provides with experiment services, the Wechat mini program is where the user interacts with the system. The software primarily implements user system, device management, experiment algorithm downloads, and remote experiment monitoring functionalities. Correspondingly, the Wechat mini program should provide pages and views where users can access these experiment services.

2.2.1 User System

Every user of the NCSLab mini-program needs to log in to the platform using their own account. As a mobile version of NCSLab, the NCSLab WeChat mini program should not only support logging in with NCSLab accounts that some users have created before but also enable users to log in with their WeChat accounts. When using the WeChat account for login, users do not need to enter a username and password but to click the login button. Even if it is a user's first time to use the NCSLab WeChat mini program, they should not create the account exciplitly but click the login button and the system will automatically help register the account.

2.2.2 Device Management

NCSLab's mobile platform offers a variety of experimental setups for users to choose from. But at time point, each experimental setup can only be occupied and used by one user. Therefore, the NCSLab nimi program should display the ocupation of every experiment setup and make sure that each account only does its own experiment on the plant when it is the owner of the plant.

2.2.3 Algorithm Download

Before starting the experiment, users must download an algorithm into the controller. NCSLab mini program should supply with a list of algorithms for the users to choose from, e.g PID algorithm. When downloading the algorithm, NCSLab mini program should launch a HTTP request to the experiment server in order to assign the target algorithm ID and the target plant ID.

2.2.4 Experiment Monitor

As the experiment is going on, users must be able to monitor the experiment. On one hand, the NCSLab mini program should allow the user to modify parameters of the control algorithm, such as setpoints, proportional gains, and differential coefficients of the PID algorithm. On the other hand, the NCSLab mini program should render data received from the experiment server with line charts and videos so they can see how well the algorithm is doing.

3. Design of The NCSLab mini program

3.1 Archetecture of The Mini Program

There are different ways to develop a mobile application. Currently, they can be categorized into three types: native application, web application, and hybrid application. A native application adopts the C/S (Client/Server) architecture with a client installed on the mobile phones and a server which provides with bisuness services. A web application adopts the B/S (Browser/Server) architecture and requires a browser installed on mobile phones. To develop a native app, programmers need to develop both Android and iOS versions of the app with Java or Swift languages which takes a lot of effort in to it. A web app can operate differently on various browsers and brings bad user experiences. Therefore, a hybrid application takes a share in the application programming.

WeChat mini programs are hybrid architecture applications. As shown in Fig. 2, WeChat mini-programs run within the WeChat software and can access native APIs (Application Program Interface) through the WeixinJSBridge toolkit. The key feature that enables WeChat mini-programs to be cross-platform is that the code does not directly run on different operating systems. WeChat mini-programs are developed using Wechat mini program template in WXML, WXSS, and JavaScript languages, which is recommended by the administrators and the sytax is similar to using HTML, CSS, and JavaScript in web application. Moreover a web-view component is provided to render real web pages in the devolopers toolkit. As shown in Fig. 2, a web page can also access wechat mini program APIs or native APIs through JS-SDK toolkit.

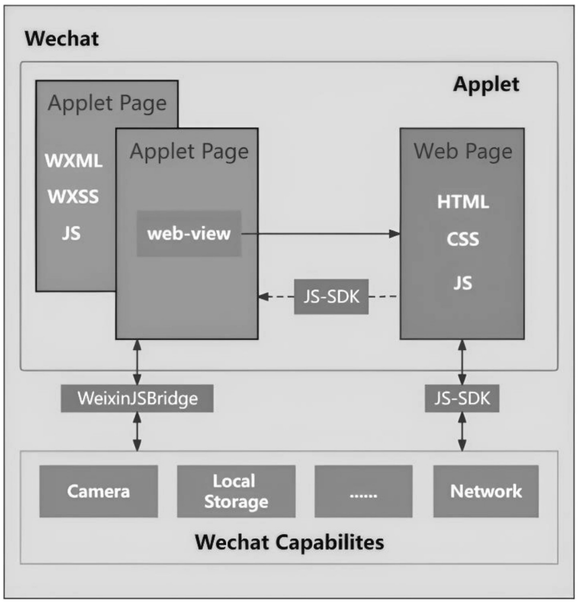

Fig. 2. Architecture of the NCSLab Mini Program.

Due to the limited development tools and resources of third party dependencies, implementing complex experiment operation functions within the WeChat mini program can be challenging. Therefore, the NCSLab mini-program utilizes web-view to render web pages in which users start and monitor an experiment. As shown in Fig. 2, the NCSLab mini-program uses the web-view component to load web pages. When the program runs, the web-view component sends a HTTP request to the server for the web page files and then start rendering. With the help of the JS-SDK toolkit, the web pages within the mini-program can access wechat mini program APIs or native APIs and complete its own task. In summary, both the applet pages and the web pages will be running in the NCSLab mini program. Before the mini program starts, it only contains part of the code and will make requests for the rest of the code which is stored in the static resource server shown in Fig. 1. This architecture helps shorten the development period and provide users with frinedly and efficient mobile platform for remote laboratory services.

3.2 Realization of The Mini Program

3.2.1 Realization of The Pages

The experimental operation interface is developed as a web page based on React. As shown in Fig. 2, the web-view component of the mini program makes HTTP requests to the NCSLab static resource server to retrieve web page files, which are then loaded and displayed by the web-view component. The experiment operation page is consisted of four tabs: Device Info Tab, Algorithm Tab, Configuration Tab, and Monitor Tab, as illustrated in Fig. 3. Device Information Tab retrieves and displays the occupation information of all copies of the current experiment plant in the form of a button list. After clicking each button, the mini program sends an HTTP request to the NCSLab server to occupy the copy of the plant. Algorithm Tab presents all available algorithms information for the current experiment plant in the format of a list. Users can select an algorithm and the mini program will send the ID of the algorithm to the experiment server via an HTTP request to initiate the download process. Configuration Tab shows a list of all available configuration files. And the Monitor Tab renders a configuration file where users can modify algorithm parameters and observe the experiment. It usually contains various components like "number display" and "line chart" to render monitoring data and a video component to display monitoring videos. The React-based web pages within the mini-program enable users to have a user-friendly and interactive experience when accessing and controlling the experimental plants remotely.

Each component is represented as an object added to the DOM (Document Object Model) tree which represents the structure of the web page. Fig. 4 shows the structure of the monitor tab, which is one of the four tabs of the experiment operation page. On the top of the structure, there is a tab container of three more tabs, each of which is shown in Fig. 8. All components that help experiment operation in the monitoring process are categorized into these three tabs, namely Inputs Tab, Data Display Tab and Video Tab. Each tab consists of a toolbar and a workspace. When users create or modify monitoring configuration files, they only need to click a button in the toolbar, and a corresponding component will be added in the workspace. Each component is wrapped in the Ant Design card component so that users can click the delete button of the card to remove the component from the workspace. The action of adding or removing a component results in the change in the component list of the web page. The web page passes the component list as a property to the tabs and the latter construct the DOM tree according to the list. In React, when a child component meets a change of the property passed by the parent com-

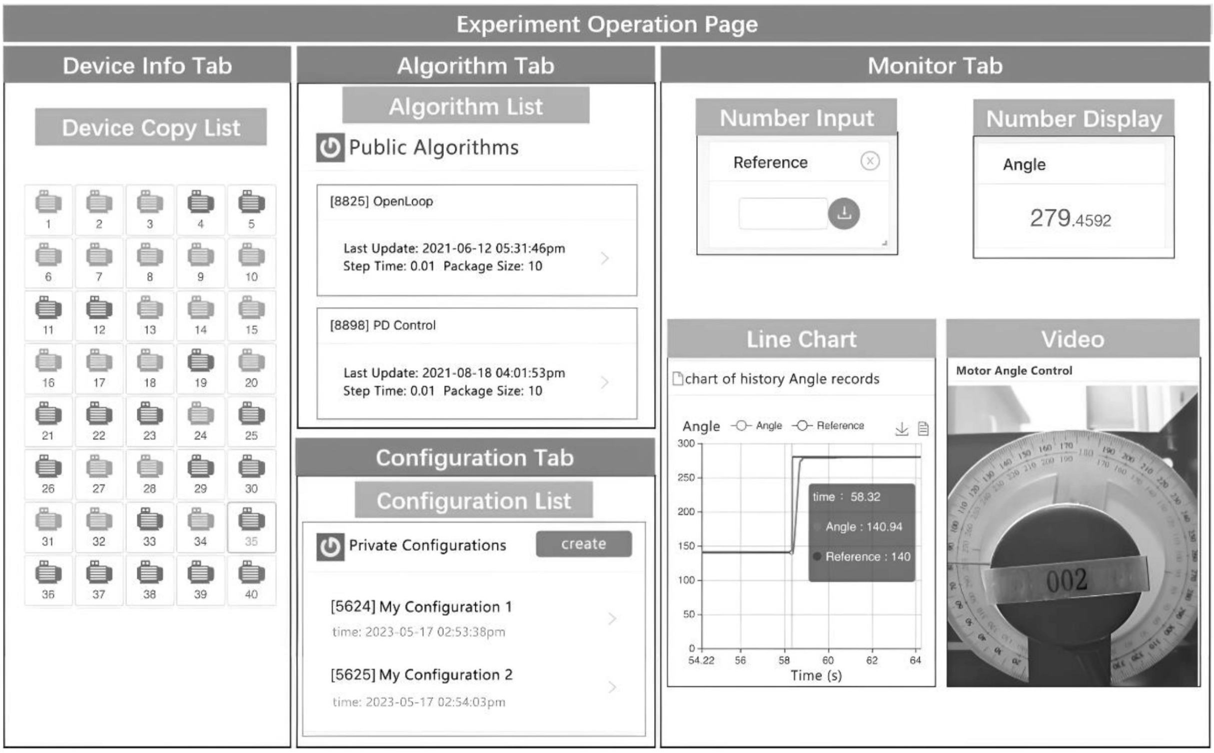

Fig. 3. Architecture of Experiment Operation Page.

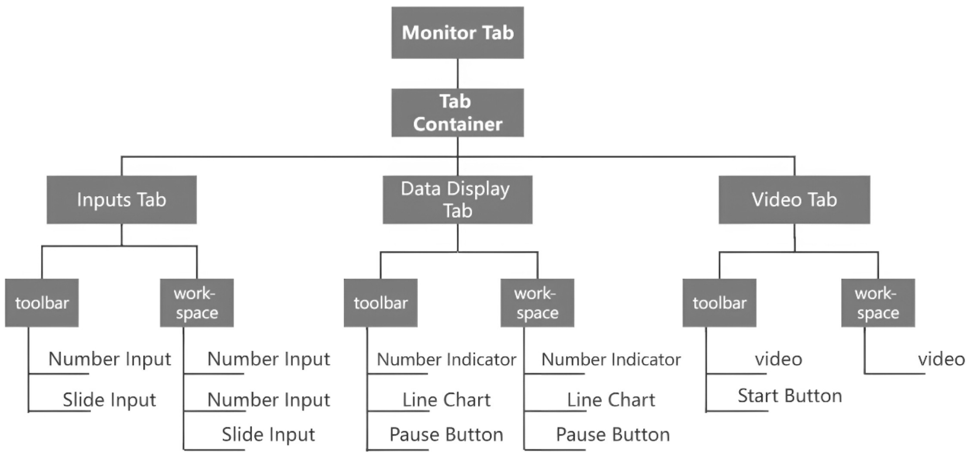

Fig. 4. DOM of Monitor Tab.

ponent, it automatically starts a re-render of the DOM tree.

The implementation of these functionalities involves using different UI components from third party resources. All four tabs mentioned above utilize UI components provided by Ant Design, namely buttons, inputs, lists, and cards. The number input component of the monitoring tab is combined by an input component to capture user input values and a download button to upload the user input value to the NCSLab server. The munber display component, on the other hand, is a disabled Ant Design input that displays realtime data of the experiment. Additionally, the monitoring tab employs the line chart component provided by Echarts to render and display historical data over a period of time. Upon clicking a specific point on the line chart, the component displays the exact numerical value at that time point. Furthermore, the video surveillance component utilizes JSMpeg to render video streams on the HTML5 page.

3.2.2 Programming of The Page

The four experiment tabs work in a uniform manner in an experiment operation to ensure that every user operates in an expected way. As depicted in Fig. 5, there exists a specific operation sequence among the operation tabs. For instance, occupying an experiment device is a prerequisite for downloading the control algorithm. Similarly, only after downloading the control algorithm can experiment monitoring be initiated. Furthermore, upon re-downloading other control algorithms, the configuration tab needs to be updated, and the monitor tab needs to be emptied. Moreover, when a user saves a configuration file on the monitor tab, the configuration tab requires an update too.

In the program, experiment operation page performs the role of the administrator. It stores all of the status, flags and other data during the experiment process and dispatches the necessary data to the tabs as properties. This approach works in two aspects. On the one hand, the tabs call methods of the parent component to update the data. On the other hand, when data transferred as properties to the tabs has changed, this event eventually cause the tabs to re-render views.

3.3 Realtime Experiment

The NCSLab mini-program supports real-time monitoring of experiments. Before starting the monitoring process, users need to download the algorithm. The communication process among the mini program client, the experiment server, and the networked controller is illustrated in Fig. 6. The mini program and the experiment server communicate through a WebSocket connnection, before which the experimental server has established a TCP connection with the networked controller in the local area network. At first, the user selects the algorithm, and the mini-program sends the corresponding algorithm ID along with the experiment plant ID to the experiment server. Subsequently, the experiment server retrieves the corresponding algorithm from the database and downloads it to the network controller of the experiment plant. Finally, the network controller returns information regarding the parameter settings of the control algorithm. This is the stage of downloading algorithm.

In the beginning of the next and last stage of experiment monitoring, users proceed to build their own monitor configuration on the Monitor Tab. This page offers a series of components for users to choose from during the monitoring process mentioned before, e.g "Line Chart" and "Numeric Display". After adding these components, users should bond each component to a specific parameter from the obtained parameter settings information as the data source of the component.

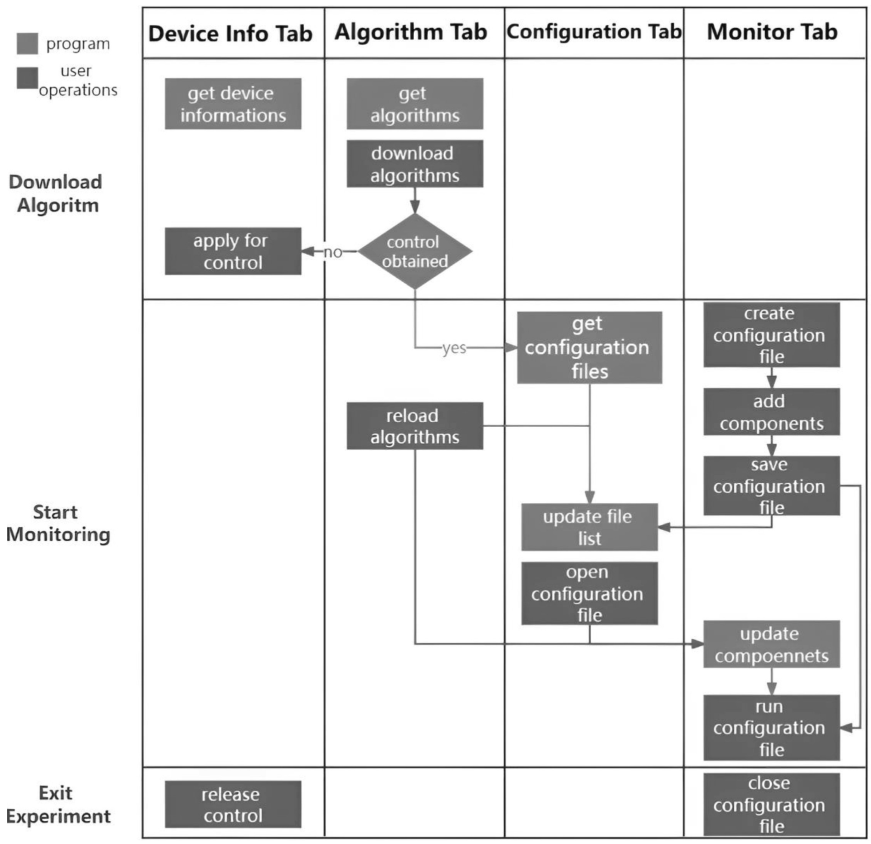

Fig. 5. Experiment Operation Process.

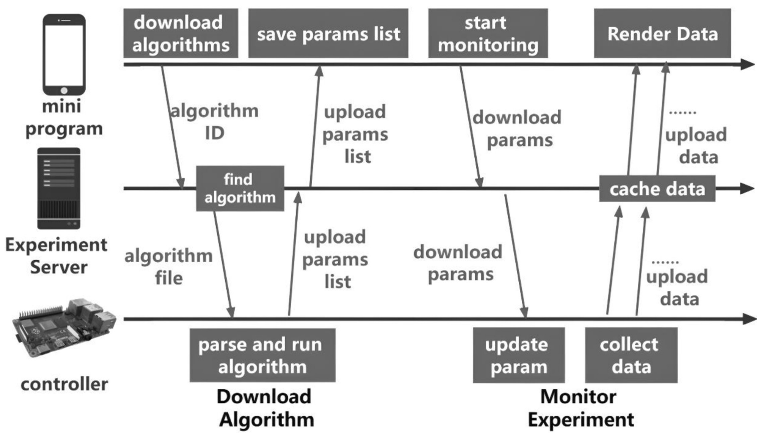

Fig. 6. Process of Communications During Experiment.

After configuring the configuration file, the user can start the experiment monitoring. The mini-program establishes a WebSocket connection with the experiment server and sends the configurations to the experiment server, which is further uploaded to the networked controller. Subsequently, the networked controller uploads the realtime data to the experiment server and the server caches and updates the experiment data series. At last the "Line Chart" and "Numeric Display" components on the page continuously update and render the data received from the experiment server. Additionally, to obtain the video stream from the webcam, the mini program establishes another WebSocket connection with the camera server. The "Video" component decodes the received video stream and renders the video pictures.

4. Case Study

During the second semester of the 2020–2021 academic year, undergraduate students who were

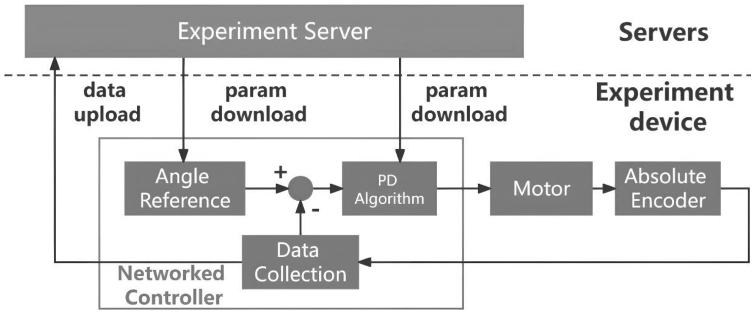

Fig. 7. Control Algorithm.

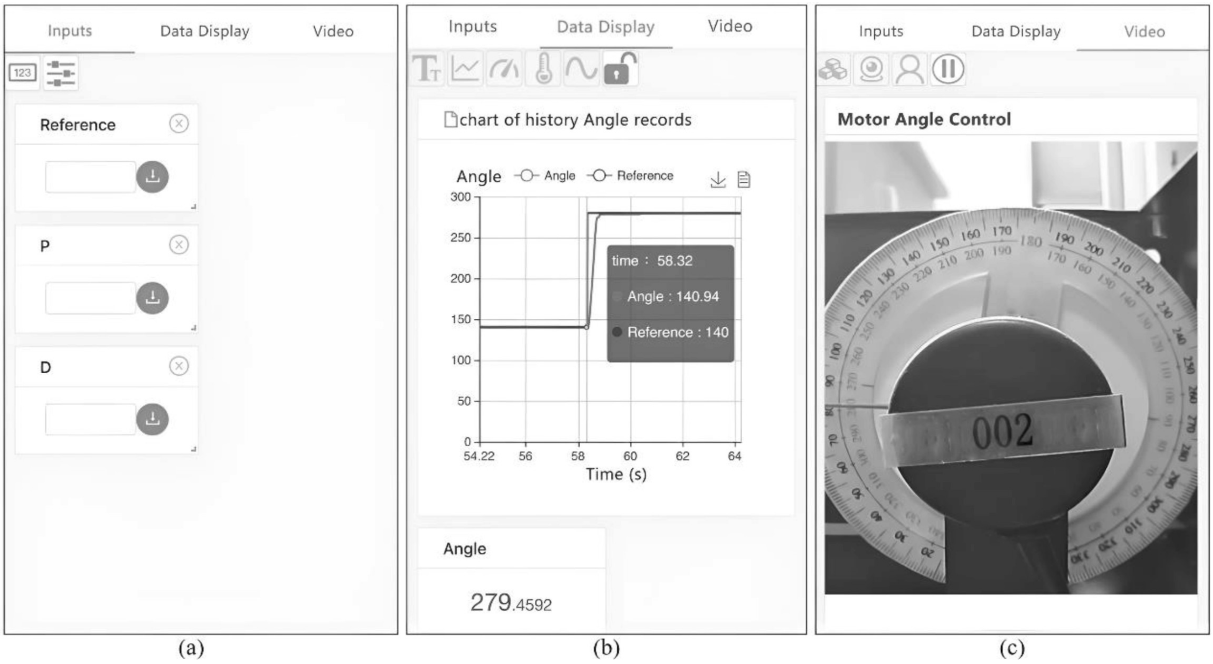

Fig. 8. Screen Shots in The Experiment. (a) Inputs Tab with input components to change parameters of the algorithm; (b) Data Display Tab with components to display motor angle value; (c) Video Tab of displaying surveillance video.

majoring in Automation Engineering at Wuhan University engaged in a remote control experiment on motor angle using a PD algorithm through the NCSLab mini program after joining in the course of "Principles of Automatic Control." In the course, students learn the fundamental knowledge of the control system, design of a controller and the analytical method for an automatic control system. An experiment was conducted based on NCSLab mini program where students were asked to complete the design of the controller in an automatic control system, namely the Motor Angle Control System. Therefore, the motor angle would rotate to the given value or the angle reference in a short time. Furthermore, the students were also asked to analyze the performance of the auto-control system they designed based on the response curves and then submit their own reports. At last, the assessment of their work involves whether or not the students successfully completed the design of the controller and the performance of the controller. The latter includes evaluating indexes of response time, percent overshoot and static accuracy.

The Motor Angle Auto-control System in the experiment is illustrated in Fig. 7. The experiment setup involves a DC motor which receives the PWM control signal generated by the controller and drives the angular rotation through the gear transmission mechanism. Simultaneously, an absolute encoder measures the rotational position of the motor. Then the networked controller collects the data and proceeds to calculate the deviation between the motor's rotational angle and the set-point. Using the PD algorithm, the controller computes the value of the control signal and subsequently converts it into PWM control signal to drive the motor's rotation.

The design of the controller involves students adjusting the proportional and derivative coefficients of the PD algorithm using the "Number Input" component in the experimental operation interface, as shown in Fig. 8(a). Students can then observe the line chart and click the "Pause" button in the monitoring workspace, as depicted in Fig. 8(b), to halt the updating of the linechart at a specific time. By swiping inside the line chart component, they can read the value of the motor angle at different time points. Therefore, students can analyze the step response curve to assess the control effectiveness. Additionally, they can use the "Video" component to observe the real-time dynamics of the device, as shown in Fig. 8(c), enhancing the overall experimental experience in a more intuitive manner.

5. Assessment

5.1 Survey

After the completion of the experiment course, students were invited to participate in a questionnaire survey to provide feedback on their experience using the NCSLab mini program. As shown in table 1, the survey contains seven questions with a 5-point Likert-style scale ("Strongly Agree" = 5, "Strongly Disagree" = 1). The firtst two questions is to measure their completeness of the experiment. The following two questions is about the user experience of the NCSLab mini program. The rest questions are about whether the app helped their learning.

Table 1. Questionnaire

Q1	I totally understand the contents of the experiment.
Q2	I finished the experiment myself and the result is satisfactory.
Q3	The NCSLab mini program has a user-friendly GUI.
Q4	I didn't encounter any exceptions during the experiments.
Q5	I gained controller design experience from the experiment.
Q6	The experiment helped me understand the course contents.
Q7	Overall, I'm satisfied with the experiment.

A total of 35 responses were collected and the result is shown in Table 2. The survey results of Q1 and Q2 indicates that over 88% of the students were able to complete all experimental operations using the platform and satisfied with the results. The average score is the same but the choices varies more when it comes to Q2. While about 80% of the students agree with the point that " The NCSLab mini program has a user-friendly GUI", some students reported encountering exceptions during the experiments which suggests the mini program needs improvement. Part of the reason is that this was the first time that the mini program came into service and some issues were neglected. And the survey results of Q5 and Q6 indicates that around 90% of the students thought that the experiment helped their learning and that they gained controller design experience from the experiment. The data also suggests more students believe they learned specific skills but less confident when it comes to the overall course content perhaps because a single experiment is not enough to cover it all. Besides, around 85% of them were also satisfied with experiment.

5.2 Discussion

The survey results show that, in general, students were satisfied with the NCSLab mini program and the experiment of Motor Angle control experiment. Perhaps one of the reasons is that students were able to do the experiment whenever and whatever they like. They can easily access the online laboratory and do experiments as long as they would like or until they are satisfied with the results. More practice would always help. While they enjoy the convenience, students also concern about the operability and stability of the app. Afterall, a new app with these advantages could stimulate their learning interest.

6. Conclusion

The first section of this paper describes the background of the NCSLab software and introduces the design of the NCSLab mini program and what kind of mobile services can be provided in the current context of Mobile technology. Based on this, the

Table 2. Questionnaire Results

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Average	S.D.
Q1	11	22	2	0	0	4.257	0.553
Q2	13	18	4	0	0	4.257	0.648
Q3	10	18	7	0	0	4.086	0.692
Q4	8	16	8	1	2	3.771	1.017
Q5	14	18	3	0	0	4.314	0.622
Q6	12	19	4	0	0	4.229	0.636
Q7	13	17	5	0	0	4.229	0.680

second section introduces the technical details of implementing such remote experiment service: the hybrid architecture of the mini program, the use of web technologies inside of it and the DOM tree of the web page. In addition, this section also explains the process of the communications among the mini program, experiment server and the networked controller. In the third section, with the help of the case of motor angle PD control, the ability of NCSLab mini program to provide remote experimental services is verified in concrete practice.

With the rapid development of Mobile technology, NCSLab mini program provides Mobile services for current college experiment education. This remote experimental service is not intended to subvert the traditional experimental teaching service. On the contrary, it is committed to providing the same learning experience as traditional teaching process through mobile technologies. The implementation of NCSLab mini program and the utilization of the program in the experiment course proves the point.

References

1. Ozhan Tingoy and Sabri Serkan Gulluoglu, The future of learning: d-learning to e-learning to m-learning, *ICERI2009 Proceedings*, Madrid, Spain, 16–18 November 2009, pp. 4066–4073, 2009.
2. Jeanne Lam, Jane Yau and Simon K. S. Cheung, A Review of Mobile Learning in the Mobile Age, ICHL 2010: Hybrid Learning, Beijing, China, 16–18 August 2010, pp. 306–315, 2010.
3. Amel Douar, Djoudi Mahieddine, Saad Harous and Alti Adel, 3DVL@ ES: A 3D Web-Based Virtual Laboratory for Collaborative Learning in Experimental Science Practical Work, *International Journal of e-Collaboration*, **19**(1), pp. 1–26, 2023.
4. Andrej Trost and Andrej Zemva, A web-based tool for learning digital circuit high-level modeling, *The International Journal of Engineering Education*, **35**(4) pp. 1224–1237, 2019.
5. Sukriti Dhang and Cheruvu S. Kumar, Efficient web-based simulation on analog electronics circuits laboratory, *Computer Applications in Engineering Education*, **31**(3), pp. 777–788, 2023.
6. Marjan Golob and Bozidar Bratina, Web-based control and process automation education and industry 4.0, *The International Journal of Engineering Education*, **34**(4) pp. 1199–1211, 2018.
7. Yuliang Qiao, Guo-Ping Liu, Geng Zheng and Wenshan Hu, NCSLab: A Web-Based Global-Scale Control Laboratory With Rich Interactive Features, *IEEE Transactions on Industrial Electronics*, **57**(10), pp. 3253–3265, 2010.
8. Lei Cao, Guo-Ping Liu, Wenshan Hu and Jahan Zaib Bhatti, Design and Implementation of an Android-based Control Laboratory, *39th Chinese Control Conference (CCC)*, Shenyang, China, 27–29 July 2020, pp. 4450–4455, 2020.
9. Anzhelika Parkhomenko, Myroslav Zadoian, Aleksandr Sokolyanskii, Artem Tulenkov, Yaroslav Zalyubovskiy, Andriy Parkhomenko, Heinz-Dietrich Wuttke and Karsten Henke, Modern Mobile Interface for Remote Laboratory Control, *Proceedings of the 23rd International Conference on Interactive Collaborative Learning*, Tallinn, Estonia, 23–25 September 2020, pp. 584–592, 2021.
10. F. Yudi Limpraptono, Eko Nurcahyo and Ahmad Faisol, The development of electronics telecommunication remote laboratory architecture based on mobile devices, *International Journal of Online and Biomedical Engineering*, **17**(03), pp. 26–26, 2021.
11. S. C. Ng, Andrew K. Lui and S. H. Ngao, An Interactive Mobile Learning Platform for Teaching and Learning Chinese Language in Secondary School Environment, ICT 2013: Knowledge Sharing through Technology, Hong Kong, China, 10–11 July 2013, pp. 135–147, 2013.
12. Xingwei Zhou, Wenshan Hu and Guo-Ping. Liu, React-Native Based Mobile App for Online Experimentation, *39th Chinese Control Conference (CCC)*, Shenyang, China, 27–29 July 2020, pp. 4400–4405, 2020.
13. Yuliang Qiao, Guo-Ping Liu, Geng Zheng and Wenshan Hu, A solution of web-based real-time control laboratories, *2009 European Control Conference (ECC)*, Budapest, Hungary, 23–26 August 2009, pp. 4605–4610, 2009.
14. Zhongcheng Lei, Hong Zhou, Shengwang Ye, Wenshan Hu and Guo-Ping Liu, Cost-Effective Server-side Re-deployment for Web-based Online Laboratories Using NGINX Reverse Proxy, *IFAC-PapersOnLine*, **53**(2), pp. 17204–17209, 2020.
15. Shiqi Guan, Wenshan Hu and Hong Zhou, Front-end and Back-end Separation – React Based Framework for Networked Remote Control Laboratory, *37th Chinese Control Conference (CCC)*, Wuhan, China, 25–27 July, pp. 6314–6319, 2018.
16. Liwei Xue, Wenshan Hu, Guo-Ping Liu and Hong Zhou, Simulink-Based Online Algorithm Design Interface for Web-Based Control Laboratory, *39th Chinese Control Conference (CCC)*, Shenyang, China, 27–29 July, pp. 4412–4417, 2020.

Kezhi Li received the BS degree in automation from Wuhan University, Wuhan, China, in 2021. He is currently working towards the PhD degree in control science and engineering in Wuhan University. His current research interests include networked control systems, web-based remote and virtual laboratories.

Guo-Ping Liu received the BEng and MEng degrees in automation from the Central South University, Changsha, China, in 1982 and 1985, respectively, and the PhD degree in control engineering from the University of Manchester, Manchester, U.K. in 1992. He is a professor with South University of Science and Technology, Shenzhen, China. His current research interests include networked multi-agent control systems, nonlinear system identification and control, advanced control of industrial systems and multiobjective optimization and control. He has authored/co-authored over 400 journal papers and 10 books on control systems. Dr. Liu was the Editor-in-Chief of the International Journal of Automation and Computing. He is a member of the Academy of Europe and a Fellow of IEEE, IET and CAA.

Wenshan Hu received the BS and MSc degrees in control theory and applications from Wuhan University, Wuhan, China, in 2002 and 2004, respectively, and the PhD degree in control engineering from the university of Glamorgan, Pontypridd, U.K. in 2008. He is currently a professor in the Department of Artificial Intelligence and Automation, Wuhan University. His research interests include network-based control laboratories and wireless power transfer.