CLAIMS

1. A teaching system, comprising a background service module, wherein

the background service module is configured to: receive an experimental operation instruction sent by a front-end application module, wherein the experimental operation instruction is an operation instruction for performing a step of a target experiment in a curriculum package, the curriculum package comprises one or more programming experiments, and the target experiment is any one of the one or more programming experiments;

execute the experimental operation instruction based on a running dependency package corresponding to the target experiment, to obtain an execution result; and

send the execution result to the front-end application module.

2. A teaching system, comprising a front-end application module, wherein

the front-end application module is configured to:

receive an experimental operation instruction, and send the experimental operation instruction to a background service module, wherein the experimental operation instruction is an operation instruction for performing a step of a target experiment in a curriculum package, the curriculum package comprises one or more programming experiments, and the target experiment is any one of the one or more programming experiments; and

receive and display an execution result corresponding to the experimental operation instruction.

3. A teaching method, applied to a background service module, comprising:

receiving an experimental operation instruction sent by a front-end application module, wherein the experimental operation instruction is an operation instruction for performing a step of a target experiment in a curriculum package, the curriculum package comprises one or more programming experiments, and the target experiment is any one of the one or more programming experiments;

executing the experimental operation instruction based on a running dependency package corresponding to the target experiment, to obtain an execution result; and

sending the execution result to the front-end application module.

4. The method according to claim 3, wherein the curriculum package further comprises curriculum content and interface document(s) corresponding to the one or more programming experiments; and

the method further comprises:

upon receiving the experimental operation instruction, determining, based on the interface documents, the running dependency package corresponding to the target experiment from among the running dependency packages corresponding to the one or more programming experiments.

5. The method according to claim 3, wherein the background service module comprises a CPU resource and/or a GPU resource; and

the executing the experimental operation instruction based on the running dependency package to obtain an execution result includes:

determining, based on the running status of the CPU resource and/or the GPU resource, a target CPU and/or a target GPU configured to execute the experimental operation instruction; and

executing a target instruction in the running dependency package by using the target CPU and/or the target GPU, to obtain the execution result, wherein the target instruction is an instruction in the running dependency package that is corresponding to the experimental operation instruction.

6. The method according to claim 3, wherein the curriculum package comprises teaching content and programming support content, the teaching content comprises the programming experiment, and the programming support content comprises the running dependency package; and

the method further comprises:

receiving an editing instruction for editing the teaching content and/or the programming support content; and

editing the teaching content and the programming support content based on the editing instruction, to generate the curriculum package.

7. The method according to claim 6, wherein the receiving an editing instruction for editing the teaching content and/or the programming support content comprises:

receiving a first editing instruction from a first front-end application module to edit the teaching content; and/or receiving a second editing instruction from a second front-end application module to edit the programming support content; and

editing the teaching content based on the first editing instruction to generate the curriculum package; and/or editing the programming support content based on the second editing instruction to generate the curriculum package.

8. The method according to claim 7, further comprising:

in response to an invitation instruction received from the first front-end application module, sending an invitation request to the second front-end module, to invite the second front-end application module to edit the curriculum package, wherein the invitation instruction comprises an identifier of the second front-end application module; or

in response to an invitation instruction received from the second front-end application module, sending an invitation request to the first front-end module, to invite the first front-end application module to edit the curriculum package, wherein the invitation instruction comprises an identifier of the first front-end application module.

9. The method according to claim 7, further comprising:

setting a permission of the first front-end application module and a permission of the second front-end application module in a way that the permission of the first front-end application module is different from the permission of the second front-end application module.

10. A teaching method, applied to a front-end application module, comprising:

receiving an experimental operation instruction, and sending the experimental operation instruction to a background service module, wherein the experimental operation instruction is an operation instruction for performing a step of a target experiment in a curriculum package, the curriculum package comprises one or more programming experiments, and the target experiment is any one of the one or more programming experiments; and

receiving and displaying an execution result corresponding to the experimental operation instruction.

11. The method according to claim 10, further comprising:

displaying profile information of the curriculum package.

12. The method according to claim 10, further comprising:

upon receiving a selection operation on the target experiment in the curriculum package, displaying at least one of profile information, description information, step information, or summary information of the target experiment.

13. The method according to claim 10, further comprising:

importing a preprocessing instruction corresponding to the target experiment.

14. The method according to claim 10, further comprising:

displaying a user interface corresponding to profile information of the curriculum package, so as to receive the experimental operation instruction in the user interface.

15. The method according to claim 10, wherein the curriculum package comprises teaching content and programming support content, the teaching content comprises the programming experiment, and the programming support content comprises a running dependency package of the programming experiment; and

the method further comprises:

sending an editing instruction for editing the teaching content and/or the programming support content to the background service module.

16. The method according to claim 15, wherein the sending an editing instruction for editing the teaching content and/or the programming support content to the background service module comprises:

sending, to the background service module, a first editing instruction to edit the teaching content; and/or

sending, to the background service module, a second editing instruction to edit the programming support content.

17. The method according to claim 15, further comprising:

sending an invitation instruction to the background service module, so that the background service module invites another front-end application module to edit the curriculum package, wherein the invitation instruction comprises an identifier of the another front-end application module.

18. An electronic device, comprising:

a processor, and

a memory configured to store an instruction executable by the processor,

wherein the processor is configured to call the instruction stored in the memory to perform the method according to any one of claims 3 to 17.

19. A computer-readable storage medium, storing a computer program instruction, wherein the method according to any one of claims 3 to 17 is implemented when the processor executes the computer program instruction.

20. A computer program product, comprising a computer program instruction, wherein the method according to any one of claims 3 to 17 is implemented when the processor executes the computer program instruction.