

基于CodeMirror的Web在线编程实训模块设计与实现

黄小冬

(重庆工商职业学院, 重庆 合川 401520)

摘要:“互联网+”时代,编程能力的培养将是技术技能人才培养的重要范畴。MOOCs在Web编程课程资源的传播上发挥着巨大作用,但其对编程的实训过程支持欠佳。本文基于CodeMirror编辑器,探讨Web在线编程实训模块设计及实现语法高亮与语法检查、可编辑区域的定义与使用、实训步骤的拆分及实训结果检测规则的方式,以弥补这一不足。

关键词: CodeMirror; Web编程; 在线实训; 设计与实现

中图分类号: TP311 **文献标识码:** A

Design and Implementation of Web Online Programming Training Module Based on CodeMirror

HUANG Xiaodong

(Chongqing Technology & Business Institute, Hechuan 401520, China)

Abstract: In the Internet+ era, programming skill is an important area for the cultivation of technical personnel in the future. MOOCs plays a huge role in the dissemination of Web programming curriculum resources, but lacks support for programming training process. Based on the CodeMirror editor, this paper explores the design of Web online programming training module and how to implement syntax highlighting and grammar checking, definition and use of editable areas, splitting of training steps and training results detection rules to solve the problem.

Keywords: CodeMirror; Web programming; online training; design and implementation

1 引言(Introduction)

现如今,各行各业都在探索与互联网的深度融合,对技术技能型人才的需求也在发生着变化。面向未来,技术技能型人才尤其需要“互联网+”思维与能力,具有编程能力将更具竞争力,这已成为高等教育界乃至K12教育的共识^[1]。Web编程以其入门较易、使用面广的特点,尤其适合“互联网+”技术技能型人才培养。得益于MOOCs的流行^[2],Web编程的课程资源丰富且较容易获得。

但是,这些MOOCs对编程实训的支持并不好,课程的实训部分一般以离线练习、常规在线题库如选择题、问答题的方式进行。编程作为一项技能,学习者须通过动手实践才能深入掌握。常规在线题库不能满足这一需求,而离线练习又使学习者难以得到及时有效的反馈。对于相当多的学习者来说,初学编程被认为是一项艰巨而具有挑战性的任务,编

程类型网络课程的完课率也较低^[3]。Web编程类的MOOCs需要实现对学生实训环节的有效支持。一般而言,MOOCs系统对用户系统、课程信息、章节知识点结构的定义均有较好支持与实现。基于此,本文将专门探讨如何设计与实现Web在线编程实训模块。

2 CodeMirror编辑器(CodeMirror editor)

在线编程即是通过浏览器进行代码的编辑、调试甚至部署^[4]。编辑器不仅仅是一个内容编辑框,而是实现Web编程在线实训过程性支持的关键。编辑器需要具有桌面IDE编辑器的部分功能,如代码高亮、语法提示等;还要具有可拓展性,提供接口,从而实现输入范围约束、输入内容的获取等。

满足这些要求的在线代码编辑器目前主流有CodeMirror、Ace、Quill、Medium-editor、Draft.js。其中CodeMirror是功能强大的开源软件,尤其适合Web在线编程

实训，见表1。

表1 CodeMirror编辑器
Tab.1 CodeMirror editor

功能需求	CodeMirror	
	实现方式	具体细节
根据语言类型代码高亮	Mode	支持129种语言类型
语法检查、提示	Hint	Html-hint CSS-hint Javascript-hint
可输入区域限制	Bookmark	采用Bookmark标记可编辑区域

3 Web在线编程实训模块设计(Web Online programming training module design)

3.1 数据表设计

采用MongoDB数据库进行数据表设计。MongoDB属于NoSQL数据库，其管理技术的特点主要是基于“键值对”(key—value)数据模型来进行数据管理，它打破了传统的关系模型，以一种模式自由的方式存储数据^[5]。本文将每个实训题(Practice)存储为一个MongoDB的BSON文档，包含实训题ID(practiceId)、实训题名称(practiceName)、课程知识章节ID(sectionId)、内容说明(content)、初始代码包(codePackage)、代码文件的可编辑区域设置(config)、结果检查规则(node)。其中，config为Object类型，以文件全路径名称(“.”替换为“#”)为键，以文件的可编辑方式(editable)、可编辑区域的行数范围(block)为值。为更好支持学习者自主编程实训，参照支架式教学理念，支持步骤分解实训目标，实现细粒度的实训支持。Node为实训的每一步定义结果检查规则。如图1所示。

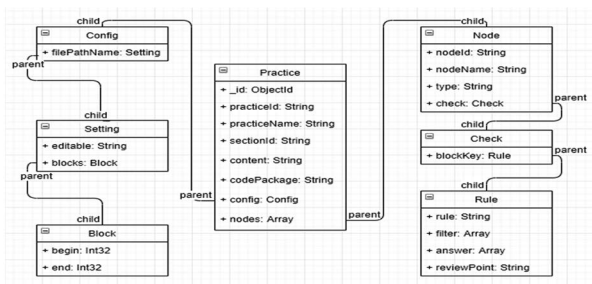


图1 数据表设计

Fig.1 Data table design

3.2 Web在线编程实训模块功能设计

从教师实训题编辑、学生在线实训两方面对Web在线编程实训模块进行功能设计。如图2所示，教师实训编辑功能主要包括实训题基本信息管理、实训题代码管理、可编辑区域管理、结果检查规则管理。创建新的实训题后，可将包括各种库的代码打包上传，Web在线编程实训模块以此生成实训代码目录。教师可创建、删除目录和文件，或更新部分文件的代码。设置可编辑区域的起始行、结束行。创建实训步骤，每步绑定可编辑区域，设置结果检查规则：人工检查，需设置审核说明；自动检查，需设置匹配方式、过滤字符、答案。

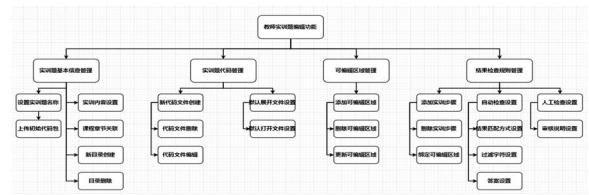


图2 Web在线编程实训模块功能

Fig.2 Web online programming training module features

学生在线实训功能主要包括代码编辑、代码保存、步骤结果验证。学生进行Web在线编程实训过程中，按照实训题的步骤分隔，在限定可编辑区域进行代码的编辑、保存，提交服务器端进行结果验证。

4 Web在线编程实训模块实现(Web online programming training module implementation)

4.1 接口的实现

Web在线编程模块采用Vue+Axios实现与RESTful API接口便捷的接入。REST风格的Web API接口，凭借轻量级、易访问、可扩展性等特性，获得了Web开发领域广泛青睐^[6]。采用PHP7语言，使用Phalcon Mirco框架实现实训题Practice的数据RESTful API接口。接口的安全性由MOOCs的Oauth2机制或加密串提供。用户数据、课程章节数据均采用MOOCs课程系统自身接口，在Web在线编程实训模块接入的过程中，仅需提供数据接口、做好数据字段对应即可。

4.2 语法高亮与语法检查

Web在线编程环境中采用CodeMirror的Mode加载方式实现不同语言的语法高亮^[7]。预先加载mode下htmlmixed、css、javascript、vue、nginx、php等支持文件，依据打开文件的拓展名动态改变CodeMirror编辑器的mode属性，从而实现特定语言的语法高亮显示。语法检查则通过加载html-hint.js、css-hint.js、javascript-hint.js等实现。

4.3 实训题的创建

为了便于实训题的归类与管理维护，需要将其与课程知识章节节点绑定。根据获取到的课程知识节点章节接口数据生成目录树结构，限制在非目录节点上方可创建新的实训题，新创建的实训题将自动绑定章节节点，效果如图3所示。



图3 实训题的创建

Fig.3 Creation of practical training questions

4.4 Web在线编程实训的实现

(1)代码初始化与维护

实训题创建后，需上传代码包。采取压缩包的方式进行初始化，以支持包含大量文件如前端库的实训题。上传解压后服务器端生成实训题工作目录。教师可编辑代码文件，创建、删除目录与文件。

(2) 可编辑区域设置

可编辑区域由开始行、结束行组成。开始行数为数值, 结束行数可为数值或缺省(表示直至文末)。可编辑区域的范围不允许重叠, 且仅允许一个区域缺省结束行。学生编程实训过程中, 代码行数可能发生变化。因此, 可编辑区域为一个动态范围, 教师显式设置范围、教师或学生进行代码编辑时均会发生可编辑区域变化。采用CodeMirror的Bookmark对可编辑区域的范围进行标记^[7], 并结合CodeMirror事件实现对可编辑区域的更新。Bookmark的构造函数如下:

```
doc.setBookmark(pos: {line, ch}, ?options: object)
→ TextMarker
```

每个区域的开始行、结束行均在ch: 0位置创建一个Bookmark。最后一个可编辑区域若直至文末, 则不创建Bookmark。在CodeMirror编辑器的ready、refresh事件回调中执行Bookmark的创建。change事件回调中重新获取代码编辑后的Bookmark位置并记录, 代码如下:

```
let marks=cm.getDoc().getAllMarks()
let zone=[]
for (var i=0;i<marks.length;i=i+2){
let begin=marks[i].find().line
let end=cm.getDoc().lineCount()+1
if(marks[i+1]&&marks[i+1]!==0&&typeof(marks[i+1])!
== 'undefined'){
end=marks[i+1].find().line
}
zone.push([begin,end])
}
```

此外, 为了区别可编辑区域, 在CodeMirror的renderLine事件回调中对每行进行判断。如果该行处于某个可编辑区域中, 则添加CSS样式。教师可在代码文件任意位置编辑, 学生则只能在可编辑区域内进行代码编辑。在CodeMirror的光标激活cursorActivity事件回调中, 根据光标点击行改变代码文档的读写属性。效果见图4。



图4 可编辑区域的设置

Fig.4 Editable zones setup

(3) 结果检测规则的实现

结果检测按步骤进行。每个步骤可绑定多个可编辑区域, 见图5。可编辑区域的结果会在change事件中自动获取。在自动检测方式中, 服务器端将各区域的代码按定义的结果检测规则进行验证是否满足。人工检测方式, 则由教师完成

代码审阅。



图5 实训步骤绑定可编辑区域

Fig.5 Training steps bound with editable zones

自动检测方式, 需设置结果匹配方式: 相等匹配或正则匹配, 分别满足简单验证与复杂验收需求。正则表达式使用单个字符串可以描述一系列满足某个句法规则的字符串集合, 因此其语义表达能力和灵活性远远高于精确字符串^[8]。过滤字符串可以自定义或选择常见的空格、回车、分号。答案可以添加多个, 满足其一即验证为通过。

5 结论(Conclusion)

本文基于CodeMirror编辑器设计Web在线编程实训模块, 实现对HTML、CSS、JS等多种Web编程语言的语法高亮与语法检查; 设计并实现可编辑区域、结果检测规则的自定义与限定, 根据实训目标细分实训步骤实现学生自主在线编程实训的细粒度支持。设计并实现的Web在线编程模块将有效弥补MOOCs平台普遍存在编程实训过程性支持不足的问题, 对相关系统的设计与开发具有参考价值。

参考文献(References)

- [1] Fatih Soykan, Sezer Kanbul. Analysing K12 Students' Self-Efficacy Regarding Coding Education[J]. Association for Information Communication Technology Education and Science, 2018, 2(1): 182-187.
- [2] Jos é Azevedo. MOOC Success Factors: Proposal of an Analysis Framework[J]. Journal of Information Technology Education: Innovations in Practice, 2017, 16(1): 233-251.
- [3] Sohail Iqbal Malik. Comparison of Traditional and ADRI Based Teaching Approaches in an Introductory Programming Course[J]. Journal of Information Technology Education: Research, 2016, 16(1): 265-283.
- [4] 胡星, 王泽瑞, 李烁, 等. POP: 一个基于微服务架构的在线编程系统[J]. 计算机科学, 2017, 44(04): 8-11.
- [5] 李常青, 谷建华. 一种基于ANTLR的MongoDB数据库SQL转化模型[J]. 西北工业大学学报, 2017, 35(01): 143-147.
- [6] 李莹, 柳生鹏, 赵朗, 等. 自适应RESTful Web API进化模型的研究[J]. 计算机集成制造系统, 2017, 23(05): 1020-1030.
- [7] CodeMirror. User Manual[EB/OL]. <https://codemirror.net/doc/manual.html>, 2019-01-08.
- [8] 付哲, 李军. 高性能正则表达式匹配算法综述[J]. 计算机工程与应用, 2018, 54(20): 1-13.

作者简介:

黄小冬(1986-), 男, 硕士, 讲师. 研究领域: 教育学, 教育信息化。