Pythonpad: Server-free Python Hands-on Exercise for Online Programming Classes

Jeongmin Byun
KAIST
Daejeon, South Korea
jmbyun@kaist.ac.kr

Jungkook Park
KAIST
Daejeon, South Korea
pjknkda@kaist.ac.kr

Alice Oh
KAIST
Daejeon, South Korea
alice.oh@kaist.edu

ABSTRACT

We propose Pythonpad, an open-source JavaScript library that supports web-based Python programming exercises. Unlike other standalone web-based programming tools, Pythonpad can be easily integrated into other websites. Although it runs learners' Python code in client-side web browsers, Pythonpad supports a file system, building and importing external modules, and many essential built-in Python libraries to teach basic programming concepts in CS1 classes.

CCS CONCEPTS

• **Applied computing** → **Computer-assisted instruction**; **Interactive learning environments**; *E-learning*.

KEYWORDS

programming education, interactive learning, scalable education

1 INTRODUCTION / RELATED WORK

Web-based programming tools help learners in online programming classes. The tools provide various feedback essential to the learning process, improve learner performance and self-efficacy [3]; thus, many researchers have been developing such tools during recent years. However, such tools are seldom used by educators other than developers themselves, despite the learners' benefits. According to studies, these tools are challenging to be customized and integrated into other existing systems [1]; they are too large and complicated. Well-known tools like CodeLab [1] or Python Tutor [2] are all standalone web services with server-side applications and are difficult to customize or integrate into other systems.

On the other hand, there are several web-based programming tools for teaching JavaScript packaged as JavaScript libraries. They are small and easier to use in various web environments or integrate into existing systems. For example, Khan Academy uses the

Processing.js library to provide interactive programming exercises in their CS1 class by simply embedding the modules in their existing platform. However, it is challenging to support non-JavaScript programming languages in JavaScript libraries for web browsers; thus, it is hard to find web-based programming tools packaged as a standalone JavaScript library for Python-based classes. Existing tools do not support file i/o or essential built-in libraries.

2 OVERVIEW / EVALUATION

We built Pythonpad, a web-based programming tool that supports Python, packaged as a JavaScript library. Pythonpad is designed to have the benefits of server-free web-based programming tools (e.g., highly scalable, no self-hosted server maintenance or paid subscription is needed) and supports most of the features available in server-dependent web-based programming tools.

Pythonpad runs learners' code on client-side web browsers using Brython, a Python implementation for web browsers. We added several features that are not commonly available in in-browser Python implementations but are essential in CS1 classes. Pythonpad can teach learners about file input and output, creating and using external modules, or text data processing. Furthermore, we also added three educational libraries that support RUR-PLE-like robot worlds, image processing, and vector shape drawing to show that Pythonpad can also support libraries with a graphical interface.

We ran a user study with 12 educators who experienced creating web-based programming exercises and assignments for CS classes at KAIST. We asked them to create their own website with a Pythonpad instance and then asked for their opinions on Pythonpad. Results show that Pythonpad is easy to embed and supports the essential features required in CS1 classes.

3 CONTRIBUTIONS / FUTURE WORK

The main contributions from this work are: (i) Pythonpad, an open-source JavaScript library for serving server-free web-based Python exercises, and (ii) a description of how we implemented a server-free web-based programming tool for a non-JavaScript programming language with file system support. As future work, we would like to improve Pythonpad to handle network-related features and scientific libraries in Python.

REFERENCES

[1] Valerie Barr and Deborah Trytten. 2016. Using turing's craft codelab to support CS1 students as they learn to program. *ACM Inroads* 7, 2 (2016), 67–75.

[2] Philip J Guo. 2013. Online python tutor: embeddable web-based program visualization for cs education. In *Proceeding of the 44th ACM technical symposium on Computer science education.* 579–584.

[3] Shu-Ling Wang and Pei-Yi Wu. 2008. The role of feedback and self-efficacy on web-based learning: The social cognitive perspective. *Computers & Education* 51, 4 (2008), 1589–1598.