Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems

School of Information Systems

10-2013

Creating Adaptive Quests to Support Personalized Learning Experiences when Learning Software Languages

Chris BOESCH
Singapore Management University, cboesch@smu.edu.sg

Sandra BOESCH
Pivotal Expert Pte Ltd, sandracboesch@gmail.com

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

🌀 Part of the Software Engineering Commons

Citation

CREATING ADAPTIVE QUESTS
TO SUPPORT PERSONALIZED LEARNING EXPERIENCES
WHEN LEARNING SOFTWARE LANGUAGES

Chris Boesch, PhD.
Sandra Boesch, PhD.

ABSTRACT

Over the past three years the authors have been developing and refining an online practicing platform called SingPath, which enables users to practice writing code in various software languages. The most recent feature to be released is a Quest mode that encourages users by showing short video clips each time a user solves five problems. In addition, users are able to choose whether to play through these quests on easy, medium, or hard levels of difficulty. The ability for users to customize their game play enables them to modify the difficulty of the experience and ideally self-regulate how frustrating or boring they find the practicing experience. Additionally, a drag-n-drop mode has recently been added for users that would like to practice assembling solutions in a particular programming language before moving on to attempting to write code in that language. This new drag-n-drop mode enables quests to be played on a variety of tablets as well as traditional devices with keyboards. In this talk, we will share learning experiences from launching the quest feature, how users are self-regulating the difficulty of the practice experience, and how much personalization is taking place as users of differing capability are presented problems of varying levels of difficulty.

KEYWORDS
Game-based learning, personalized learning, self-regulated learning

Introduction
When setting out to develop a more effective method to teach basic computer science, the authors were looking for innovative ways to provide additional, individualized feedback to students learning software languages such as Python, JavaScript, and Java for undergraduate university courses. The authors took the approach to enable students to practice a variety of software languages on their own by having them solve short programming problems in an online game (see Figure 1).
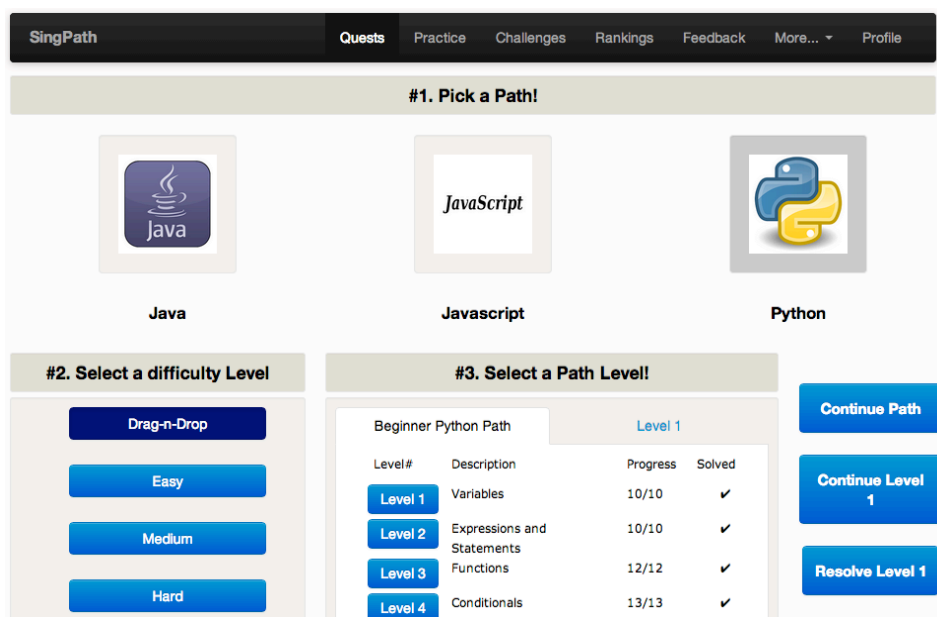

Figure 1. Selecting paths to practice.

Students were able to practice solving these problems on their own time, from their own systems, wherever they had Internet access. This method enabled the authors to provide additional feedback to students in a more real-time manner than had been previously possible with live, in-class quizzes [1] and weekly problem sets turned in as homework. Students were still assigned problems to solve as in previous terms, but by requiring students to solve all problems in an online system (see Figure 2) the authors were able to provide students with real-time feedback on their progress and at the same automatically track which students were on pace to solve all required problem prior to weekly deadlines.

Figure 2. Basic feedback and gameplay.

As more people around the world began to solve problems on the online system, two consistent categories of feedback were received. A portion of the users continued to make comments such as "These problems are too difficult" while other users would provide comments such as "I am bored. Please let me skip the easy problems." To address this feedback, the authors created an adaptive difficulty mode (see Figure 3). This enabled users to adjust the difficulty of problems to reduce the amount of boredom or frustration users might be encountering. The authors hypothesize that by enabling users to adjust the difficulty of the games they are playing, the users will be able to better balance their own boredom and frustration and increase the likelihood that they will be able to enter in to and stay in a state of learning flow while practicing (Csikszentmihalyi, 1997).

Drag and Drop

To encourage a wider audience to practice new software languages, the drag-n-drop mode was created to enable users to assemble solutions from available lines of code rather than having to type them (Boesch, & Boesch 2012). This mode provided an easier practice mode for beginners and a way to practice new software languages on mobile platforms such as tablets (see Figure 4). When users select to play a game in drag-n-drop mode, they are able to drag lines of code around and see feedback automatically rather than having to press a button to compile or test their solution.
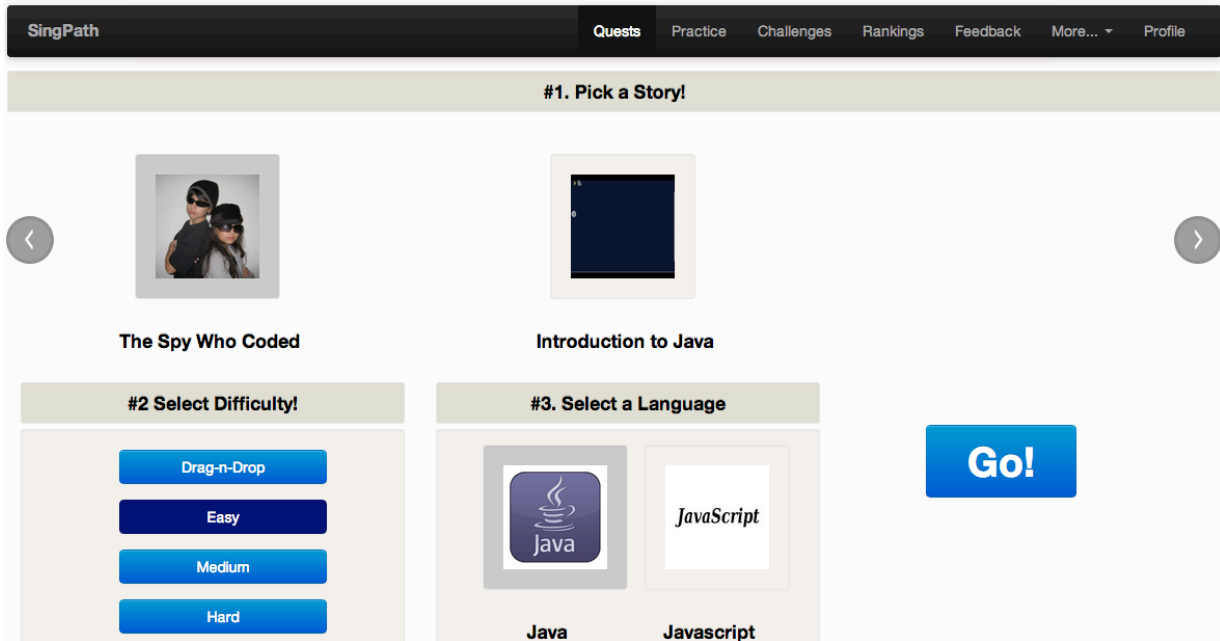
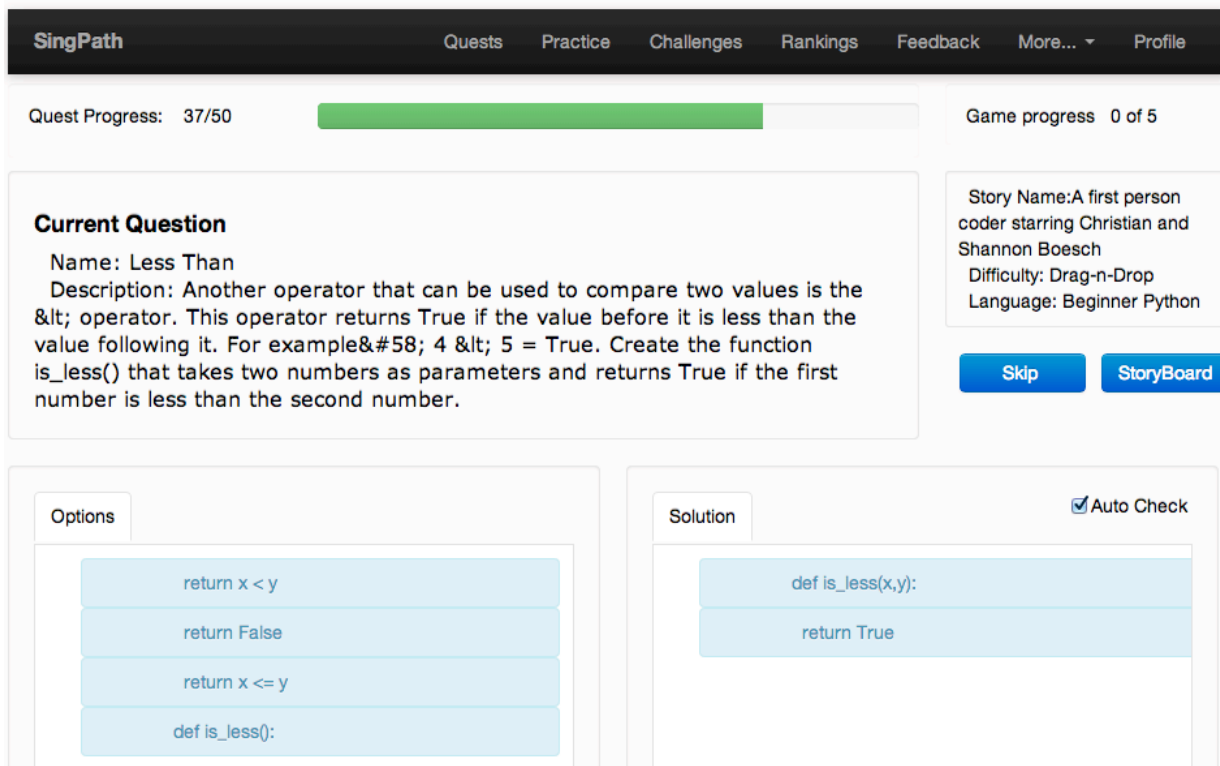Figure 3. Selecing stories, paths, and difficulty for quests.

Figure 4. Drag-n-drop gameplay.

Quests

SingPath's difficulty adaptive approach worked well for tracking and moderating problem difficulty, but it did not adequately address the varying level of interest and former knowledge that players had prior to beginning to solve problems. To address this issue, the authors implemented the ability to include a collection of short videos (see Figure 5) that are shown to the users as they solving new problems. These short videos can be used to provide relevant tutorials to users before they practice new skills, or alternatively these videos can be designed to promote curiosity and a more fun experience

intended to encourage users to practice for a longer amount of time or progress slightly further along a learning journey. The combination of working through a set of videos by working through a set of problems is referred to as a quest.
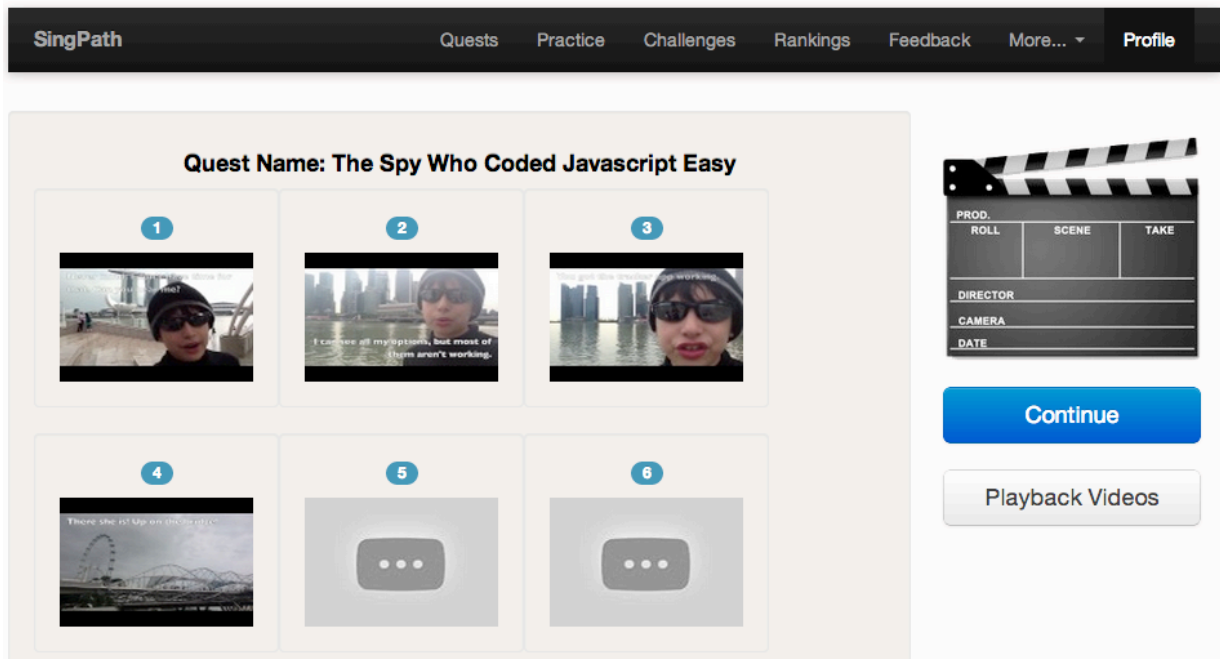

Figure 5. Unocking videos

Stories

In SingPath, an ordered collection of videos is referred to as a story. SingPath is designed so that any educator can easily create and edit their own stories simply by providing a collection of YouTube links to their videos (see Figure 6).
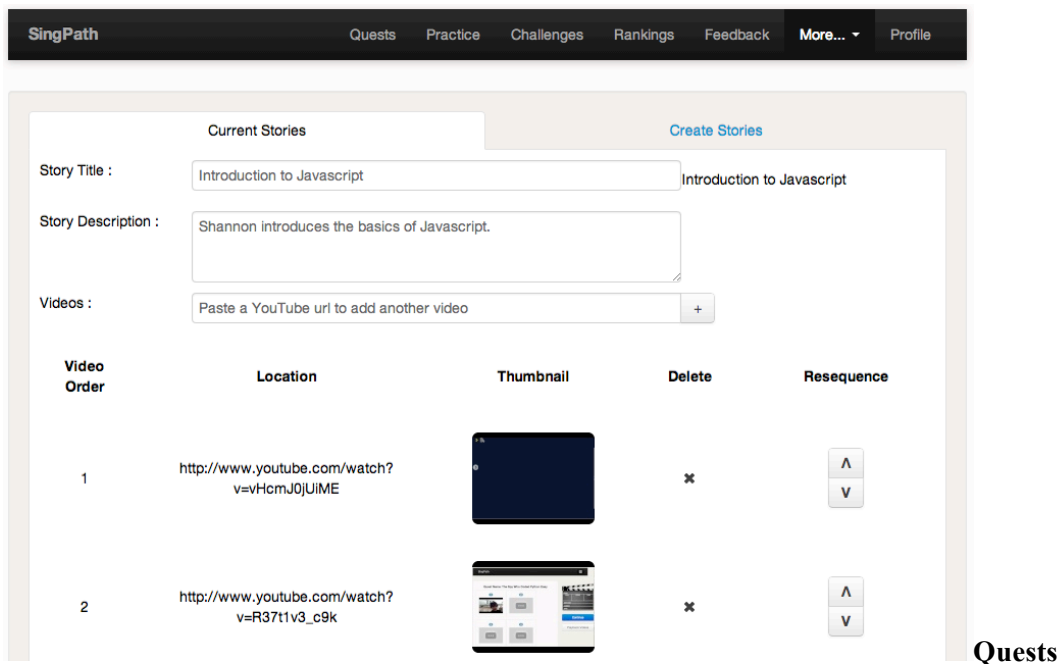

Quests

Figure 6. Creating a new story.

4

Challenges

To encourage users to practice, the authors included a variety of gamification features such as badges (see Figure 7), rankings, and completion metrics (Zichermann, & Cunningham, 2011). The authors also added support for challenges, which require users to solve a specified number of problems at a specified difficulty level before unlocking a secret message. This challenge mode enables parents, mentors, and classroom facilitators to encourage users to practice with various learning outcomes in mind (Kamei, Cook, Puthrucheary, & Starmer, 2012). With the addition of quests, educators and parents can create quest challenges (see Figure 8) that require a player to play through an entire story and collection of problems in order to unlock a secret message.

Figure 7. Badge and completion metrics

Figure 8. Creating a quest challenge.

REFERENCES

Boesch, C., & Boesch, S. (2012). Adaptive Gameplay for Programming Practice.

Boesch, C., & Boesch, S. (2011). Tournament-based Teaching. 4th Annual International Conference on Computer Science Education: Innovation and Technology (CSEIT 2012).

Boesch, C., & Steppe, K. (2011). Case Study on Using a Programming Practice Tool for Evaluating University Applicants. 3rd Annual International Conference on Computer Science Education: Innovation and Technology (CSEIT 2011).

Csikszentmihalyi, M. (1997). Finding flow: The psychology of engagement with everyday life. 1st edition. Basic Books: New York, NY

Kamei, R., Cook, S., Puthrucheary, J., & Starmer, F. (2012). 21st Century Learning in Medicine: Traditional Teaching versus Team-based Learning in Medical Science Educator. 22(2).

Michaelsen, L., Knight, A., & Fink, L. (2002). Team-Based Learning, A Transformative Use of Small Groups in College Teaching. 1st edition. Sterling, Virginia: Stylus Publishing, LLC

Zichermann, G., & Cunningham, C. (2011). Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps. Gabriel Z Inc.

AUTHOR AFFILIATION AND ADDRESS INFORMATION
Dr. Chris Boesch
Associate Professor of Information Systems (Education)
Singapore Management University
80 Stamford Road
Singapore 178902
Email: cboesch@smu.edu.sg

Dr. Sandra Boesch
Communications and Outreach Manager
Singapore Management University
80 Stamford Road
Singapore 178902
Email: sandracboesch@gmail.com