

TFPIE MOOCs discussion

TFPIE 2017, University of Kent, 22 June 2017

Notes by Elena Machkasova

Learning to write tests is important for CS education, but MOOCs don't seem to be teaching students to write tests. Are you concerned about this?

- Students use QuickCheck as they are developing their code (although using QuickCheck can be tricky for inexperienced students).
- Limiting submissions to 5 (combined with automated tests after submission) forces students to write their own tests.
- The Erlang MOOC introduced students to unit testing; one student in particular was really into testing and wrote extensively about it in the discussion forum
- In the OCaml MOOC students didn't write tests. Tests are too difficult for beginners to write
- Working within a REPL (Read-Evaluate-Print Loop) naturally encourages testing, but fails to document that it has happened, or save it for regression testing.
- The OCaml course improved testbenches during the course based on feedback.

Every student is unique and has unique needs. In the current MOOC system all students follow the same timeline and do the same exercises. However, MOOCs may provide a way to adjust the material to the student's level: give more time and support for those struggling, and provide advanced exercises for more advanced students. Is this currently done, and is there a potential for such approaches?

- Students tend to cluster by level, but there may be a large number of groups, especially in larger MOOCs.
- Coursera allows "cohorts", but the Scala MOOC hasn't tried them.
- On the social side, comment threads provide some degree of grouping

- Pairing people can be successful, but it's hard to do well; it works best when the two know each other (and each others' working styles), but this is not the norm for MOOC participants.
- Time limits on exercise submission (i.e. a week to finish) make it so that students naturally separate by levels - some finish very early, and some of those hang out in discussion forums to help others; some students submit closer to deadlines and may get help from those who finish earlier; some struggle to submit reasonably correct solutions at all. This applies even in cases where there are no hard deadlines for submission.
- It may be possible to let students self-categorize, and then use this to form groups (and move people if needed).

MOOCs have very large drop-off rates, is this a problem? Have the courses improve after the first run?

- When people were polled in the beginning about their reasons for enrolling, only 20% definitely intended to complete the course.
- **Q: is the drop rate the same for the university students taking the course for credit?** Their completion rate is better, but regular students in regular courses don't necessarily finish either.
- Attention drop rate in the classroom: logarithmic decay after a ramping up period means that after 40 minutes students' attention is way below the maximum.
- French speaking students found English lectures easier to use in recording
- It would be good to remove boring stuff from real lectures and move it to recorded lectures only: remove syntax ("Talking to the machine") and keep ideas, concepts, etc. ("Talking to humans")

How much time/effort did the recording and postproduction of lectures take?

- Erlang MOOC: very little. Most lectures were recorded by the professor himself in his office with just a computer camera and a mic. Some "Master classes" took more time and were more professionally done.
- On the other hand, for the Erlang MOOC there was substantial pre-production time: for each "step" there was a lesson plan, which laid out clearly what was in the step, the learning objectives, links to previous and subsequent steps and so on.
- Scala: a lot; more than needed. Multiple cameras, a lot of editing.
- Hand pointer on slides is nice.
- Lectures need to be close-captioned, and that could be tricky (a lot of terminology). However, CC is helpful.
- Some groups have problems with a simple laptop/mic arrangement.

Do any universities accept MOOCs for credit? Do professors direct their students to MOOCs?

- Some schools recommended MOOCs to their students.
- MOOCs cannot be accepted for credit since there is no guarantee that a student did their work on their own. [Note that FutureLearn have a small number of credit-bearing MOOCs from the University of Leeds and others.]
- Blended learning (using MOOCs as a part of an in-person course) is possible.

Is teaching MOOCs worth it? If not, what's missing?

- Doesn't necessarily pay off financially for the university, especially the first time around, but adds to reputation, benefits students. It's an investment.
- Students benefit from interactions with developers. Students were struggling compared to developers; developers were helpful.
- Interactions with developers teaches students better, more respectful professional interactions.
- Helps students engage and become more responsible independent learners.

It's been pointed out that attention span is no more than 3-8 minutes. Were all the materials really that length? There seemed to be longer (20+ minutes) materials in some MOOCs

- Online systems tend to control length and workload.
- Even 8 minutes videos were often fast-forwarded by students to the points they wanted to watch..
- There are longer blocks: 15-30 minutes. Productions shortens and streamlines (what in person would've been) longer lectures.
- Some students just download subtitles without watching the videos.
- It's helpful that videos are indexed by slides.
- It's difficult to make 3 min video, often results in an 8 min video.
- One shouldn't take the course system requirements too literally: longer videos may be ok.
- Students sometimes watch videos at a faster speed.