2016-04-16

Reconstructing design thinking and learning through code snapshots and clinical interviews

This paper combines two existing research methods—code snapshots and clinical interviews— to understand how a student’s design and and learning evolve in a university introductory programming course for engineers. Historically, clinical interviews have proven very useful for understanding the structure and dynamics of student knowledge (Sherin, 2013; Smith, diSessa, & Roschelle, 1993). But, in computing education the constraints of interviewing make it hard to resolve the fine-grain changes students may make to code over time. On the other hand, code-snapshotting systems (Blikstein et al., 2014; Jadud, 2006; Spacco et al., 2006) are very well-suited to resolve those kinds of changes, but they have historically been used to look between students for groupwise effects and correlations, rather than within students to trace the evolution of code as a design artifact. The case study we present uses a knowledge-in-pieces perspective (diSessa, 2002, 2006) to analyze more than 900 compile-time snapshots and more than 5 hours of clinical interview data and reconstruct a student’s design thinking over the trajectory of a multi-week programming project. We argue that augmenting snapshots with interview data offers substantial explanatory power for understanding how a student’s design choices reflect their design thinking and learning.