Rethinking Into Programming: A light-lifting, high-impact flipped class teaching programming through iOS app development

TREO Talk Paper

John Gallaugher
Boston College
john.gallaugher@bc.edu

John FitzGibbon
Boston College
fitzgijc@bc.edu

Shirley Cho
Boston College
shirley.cho@bc.edu

Jorge Mahecha
Boston College
mahecha@bc.edu

Abstract

Trade publications regularly list mobile application development as a skill in high demand. Student entrepreneurs and innovators are attracted to mobile platforms due to their ubiquity, utility, power, and the flexibility of the platform. Yet the path to achieving the skills necessary to build an app are filled with challenges.

From a student's perspective, most universities do not offer app development as an avenue for learning programming. It is common for students to have to complete two or more classes before enrolling in an apps class. The most popular language for a first course in programming is python, yet many students find the text-oriented nature of the language less motivating when their eventual goal may be to create consumer-reaching projects. Students also arrive with disparate experience in computing, and a conventional lecture-style class often leaves less experienced students struggling to keep pace with peers, leaving many discouraged and dissatisfied.

From the perspective of IS faculty, teaching programming is also a challenge. The amount of work to acquire and maintain skill in programing languages often does not overlap with the research focus of those in the Information Systems discipline. Platforms for mobile computing also undergo annual changes, so keeping current is difficult. Preparing lessons is time intensive, and programming classes often involve more hands-on help outside of lecture than non-tech classes.

In an effort to address these challenges, the principle author has developed materials for a flipped classroom approach to teaching new-to-programming students using the Swift programming language for iOS app development. The semester-long course delivers content through over 100 video lessons, most 30 min or less in length, where students learn skills in programming while building apps. No prior knowledge of programming is expected. Class time is spent working on exercises that apply concepts learned during the out-of-class video lectures. Course material brings students through nine apps, covering basic topics of input and calculation, through more advanced topics such as API use, JSON parsing, and creating a "full stack" social app using Google Cloud Firestore and Firebase authentication.

Materials will be demonstrated along with results of student surveys, and best practices, including the role of TAs, grading, hands-on exams, and use of an end-of-semester Student App Showcase.

References
