A Preliminary Analysis of App Inventor Blocks Programs

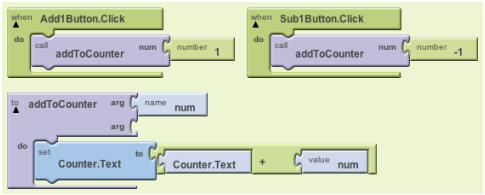
Johanna Okerlund and Franklyn Turbak Wellesley College

Project Goal

MIT App Inventor [1] is a visual programming environment for creating apps for Android mobile devices. Although over a million App Inventor users have created about 2.5 million apps, little is known about the nature of these apps and the problems encountered by the people creating them. The goal of this project is to analyze the structure of a large percentage of all App Inventor programs in order to gain insight into the effectiveness of this visual language for creating apps and for learning programming concepts.

App Inventor Background

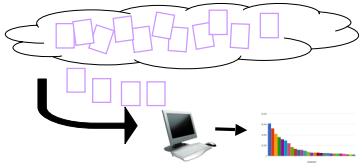
An App Inventor app is specified by a project consisting of a set of user interface components and a program that describes the behavior of these components. The program is created by connecting blocks resembling jigsaw puzzle pieces. Blocks languages like App Inventor and Scratch lower barriers for novices by eliminating or reducing many common programming errors and by providing visual guidance for choosing, assembling, and understanding program structures.



Sample App Inventor blocks program

Data Collection

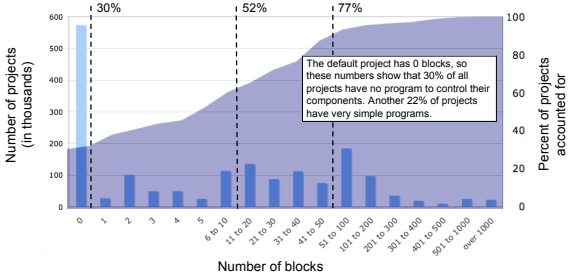
App Inventor projects are stored and backed up in the cloud



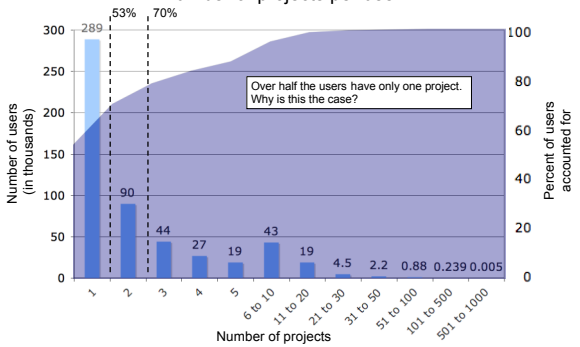
We can download and analyze the blocks from all projects that have been updated since August 2012. So far we have analyzed 1,711,167 projects from 541,671 users.

Preliminary Data Analysis

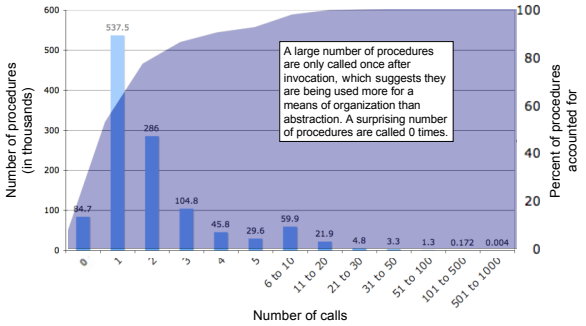
Number of blocks per project



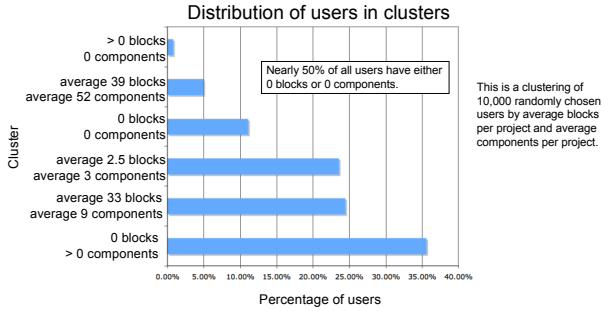
Number of projects per user



Number of calls per procedure



Distribution of users in clusters



This is a clustering of 10,000 randomly chosen users by average blocks per project and average components per project.

Future Work

Our results raise questions that demand investigation by follow-up surveys and user studies. Many users never create a block program; to what extent is this due to technical setup problems, user interface issues, and conceptual confusions? Why are there so many procedures that are declared but never called?

The recently released App Inventor 2 (AI2) simplifies the creation of blocks programs. Will statistics from AI2 indicate that the newer blocks editor is easier to use? AI2 also changes the way in which variables are declared and used. Will statistics for global variables and procedure and event parameters change?

We plan to develop notions of program sophistication in order (1) to determine the extent to which App Inventor users are learning programming concepts and (2) to provide feedback to users about their programs. These notions could be used as the basis for an automatic tutor for improving coding style.

Our work so far focuses on the structure of App Inventor programs, but it is also possible to track program execution on devices, including runtime errors. This information will help us better understand the proficiencies and confusions of users and will also allow us to provide debugging support when an error occurs.

A longer range goal is to instrument App Inventor to record fine-grained steps of program construction. This will give a more detailed narrative of how users write their programs. We expect that the high level nature of blocks programming edits will be more manageable and informative than the low-level nature of keystroke edits used in learning analytics systems for programming, such as [2].

References

[1] http://appinventor.mit.edu

[2] Chris Piech, Mehran Sahami, Daphne Koller, Steve Cooper, and Paulo Blikstein. 2012. Modeling how students learn to program. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12). ACM, New York, NY, USA, 153-160.

Funded by Brachman Hoffman Fund Faculty Small Grants Summer Research Awards and National Science Foundation Grant DUE-1226216. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.