

# Logging and Analyzing the Usage Patterns of Blockly Blocks in App Inventor



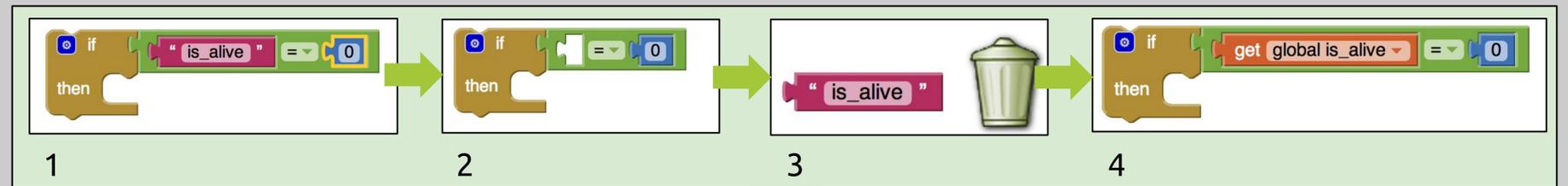
Benjamin Xie, Department of Computer Science, Massachusetts Institute of Technology

## INTRODUCTION

The objective of Blocks-Level Analytics (B.L.A.) is to generate reusable data sets of the step-by-step process behind the creation of applications with App Inventor. With this comes further insight into App Inventor as a service.

B.L.A. works by logging each action a user takes when creating Blockly script. It logs the changes in the block scripting language App Inventor uses, recording the step-by-step process of a user when developing an app. These logs enable recreation of the development process a user choose, enabling analysis into the app development process with App Inventor. In particular, this project focuses on further understanding of user errors and usability of the blocks.

B.L.A. works in the background, integrated such that users see no difference in experience when using App Inventor.



time-stamp	block	action	associated block
1	"text_box"	connect	"logic_equals"
2	"text_box"	disconnect	"logic_equals"
3	"text_box"	delete	n/a
4	"get_variable"	connect	"logic_equals"

**B.L.A. Example for Deleted Block:**  
 Here, a text block is disconnected (2) and deleted (3), replaced with reference to a variable (4) and recorded as a log. This could suggest a misconception between text and variable references, a conclusion only realized by analyzing the creation process.

## FUTURE WORK

BLA is an iterative process, consisting of creating a hypothesis, collecting the data, testing the hypothesis, applying findings, and repeating. This iterative cycles lends itself to "building up," allowing BLA to scale gracefully and be tested and improved along the way.

Given the two semester timespan, I propose two general topics to address in my SuperUROP: *Error understanding and usability groups.*

## PURPOSE

1100  
1010  
0101

**Developers** will be able to gain insight into users and how the process users go through to create program apps, enabling data-backed understanding of these patterns.



**Educators** will be able to receive reports detailing patterns across the class given a certain task or across tasks given a certain user, identifying misconceptions and allowing a more personalized learning experience.



The App Inventor **community** can expect to see an improved service as usability issues are discovered and fixed and tutorials are made more clear.

## Research Questions

### Errors

- What are the common causes to logic errors?  
To runtime errors?
- What trends cause errors?
- What trends successfully resolve errors?
- What can help users resolve different errors?
- What behaviors avoid/minimize errors?

### Usability

- How can we classify users?
- How can we predict if a new user will be retained based on initial interactions?
- Do users design iteratively? If so, how does it compare with waterfall design? (errors, app creation time)
- How do users modularize their projects? How are modularized projects different?
- What components confuse users?

## Use Cases

- **Classrooms:** Work with teacher to implement B.L.A. such that data is collected from each individual student to track learning process
- **Novices:** Track group of new users as they use App Inventor for the first time and determine how different usage patterns translate to varying results (successful apps, user retention, etc.)
- **Bug Tracking:** Determine the process that leads up to errors occurring (logic, runtime, etc.) and how users attempt to resolve errors