

# USABILITY TEST RESULTS

Kirk Thelen






















**TEAM 1** PROGRAMMING ANALOGIES

1

# APP DESCRIPTION



Misconception	Desired Knowledge	Compare	Popularity
Objects all have the same value for instance variables	Each object has its own version of the instance variables	<a href="#">Compare</a>	 4
An array's size can be changed later	Arrays have a set size determined when they are initialized	<a href="#">Compare</a>	 4
ASCII bit shift does not affect non-alphabetic characters	Shift affects any ASCII characters unless a condition states otherwise	<a href="#">Compare</a>	 2
Capitalization of variable does not matter	Variable names are case sensitive	<a href="#">Compare</a>	 1
Setting the result to double will stop integer division truncation	Floating point must be introduced to the division for floating point results to be remembered	<a href="#">Compare</a>	 1
One equal sign used to determine equality condition	Conditions of equality and assigning equality are different	<a href="#">Compare</a>	 1
Array value is its contents	The value of an array is a reference to its location in memory	<a href="#">Compare</a>	 1
Type restated on variable reuse	Declaration of type is associated with variable name	<a href="#">Compare</a>	 0
Multiple scanner creation for single input source	Only one scanner per location is necessary	<a href="#">Compare</a>	 0
Primitives and their wrapper classes are identical	Primitives and reference types are distinct	<a href="#">Compare</a>	 0
Scanner requires a delimiter to be set up	Scanner has a default delimiter on instantiation	<a href="#">Compare</a>	 0
Primitive assignments maintain references	Primitive assignments copy a value to another primitive	<a href="#">Compare</a>	 0
Code can be anywhere in a java file	The code should be contained within a class	<a href="#">Compare</a>	 0
Semicolons are not needed to end statements	Semicolons should end all regular statements	<a href="#">Compare</a>	 0
A variable can't reference itself during assignment	Operations occur before an assignment	<a href="#">Compare</a>	 0
Methods keep executing after a return statement	The method immediately ceases execution and returns any value stated in the return statement	<a href="#">Compare</a>	 0
Casting occurs after all operations in an expression	Casting will immediately cast the element after it with only parentheses having higher precedent	<a href="#">Compare</a>	 0
Methods can be defined in other methods	Methods can call other methods but each method should be defined distinctly	<a href="#">Compare</a>	 0
Header comments are not important	Header comments are vital to other programmers attempting to use your code	<a href="#">Compare</a>	 0

Web-based application for creating, sharing, and comparing analogies that explain programming concepts.

2

# TESTING PROCESS



# TESTING PROCESS

Testing was done in-person.

1. Greeting & Introduction
2. Confirmation of consent
3. Explaining App to participant
4. Participant fills out pre-test survey and questions
5. Testing begins (6 tests)
6. Participant fills out post-test survey
7. Administrator asks participant post-test interview questions
8. Thank participant & wrap up

# PRE-TEST SURVEY

- ▶ **Total Participants:** 7

- ▶ **What gender do you identify as?**

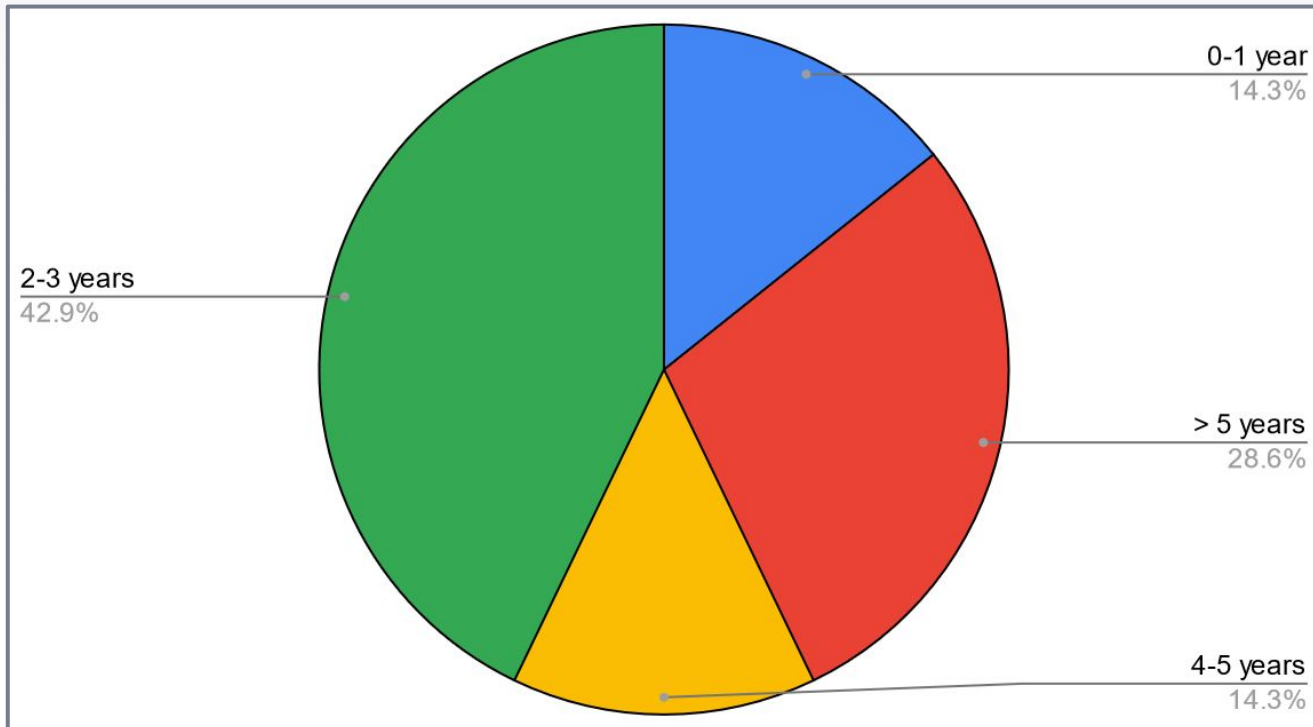
5 Men, 1 Woman, 1 Non-binary

- ▶ **What is your education level?**

1 freshman, 2 sophomores, 3 juniors, 1 graduate student

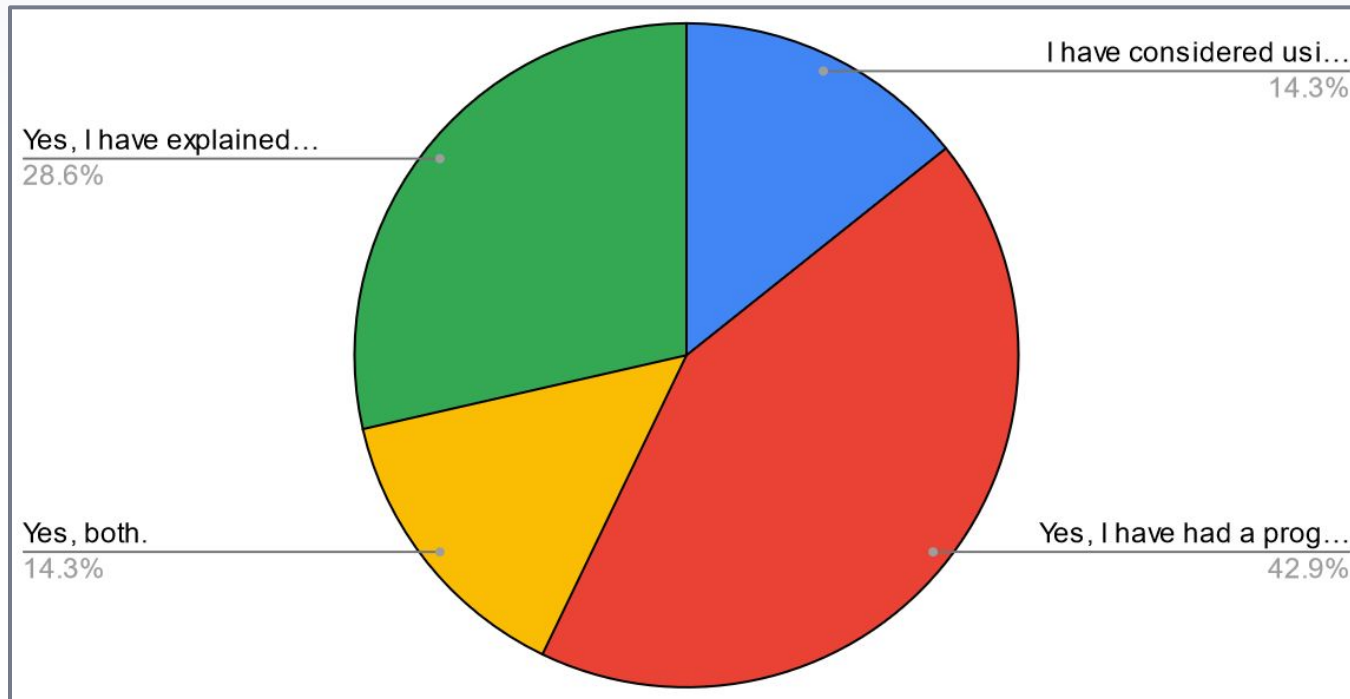
# PRE-TEST SURVEY

- ▶ What is your level of programming experience?



# PRE-TEST SURVEY

- ▶ Have you ever thought about using analogies to explain programming concepts, or have you ever had a programming concept explained to you via an analogy?





# PRE-TEST QUESTIONS

- ▶ **I am interested in the testing of this application.**  
6 agrees, 1 strongly agree
- ▶ **I understand the purpose of this application.**  
1 neutral, 3 agrees, 3 strongly agrees
- ▶ **I understand where this application may be used.**  
4 agrees, 3 strongly agrees

3

# TEST SCENARIOS



# ▶ TEST SCENARIOS: OVERVIEW

- ▶ Participants completed 6 tests.
- ▶ Each test had an expected task list.
- ▶ We recorded observations, GUI interactions, and body language.

# TEST SCENARIO SUMMARIES

**Scenario 1** Registration and Login

**Goal** Can participants register and log in to a new account?

## Results

- ▶ 7 participants found registration page
- ▶ 7 participants registered for an account
- ▶ 7 participants logged in to the account



# TEST SCENARIO SUMMARIES

**Scenario 2** Analogy Comparison

**Goal** Can participants compare two analogies?

## Results

- ▶ 7 participants could open the first analogy
- ▶ 5 participants could open the second analogy
- ▶ 5 participants closed the analogy view correctly



# TEST SCENARIO SUMMARIES

**Scenario 3** Analogy Search and Review

**Goal** Can participants search for analogies and rate/favorite them?

## Results

- ▶ 7 participants could search for an analogy
- ▶ 6 participants could select the analogy
- ▶ 4 participants could favorite an analogy
- ▶ 4 participants could return to search results



# TEST SCENARIO SUMMARIES

**Scenario 4** Analogy Creation

**Goal** Can participants create new analogies?

## Results

- ▶ 7 participants found create analogy page
- ▶ 5 participants could input data without asking questions
- ▶ 4 participants created an analogy correctly on the first try



# TEST SCENARIO SUMMARIES

**Scenario 5** Unfavorite Analogy

**Goal** Can participants unfavorite analogies?

## Results

- ▶ 7 participants found profile page
- ▶ 4\* participants could unfavorite an analogy



\*Only 4 participants successfully favorited an analogy previously



# TEST SCENARIO SUMMARIES

**Scenario 6** Analogy Deletion

**Goal** Can participants delete an analogy they've created?

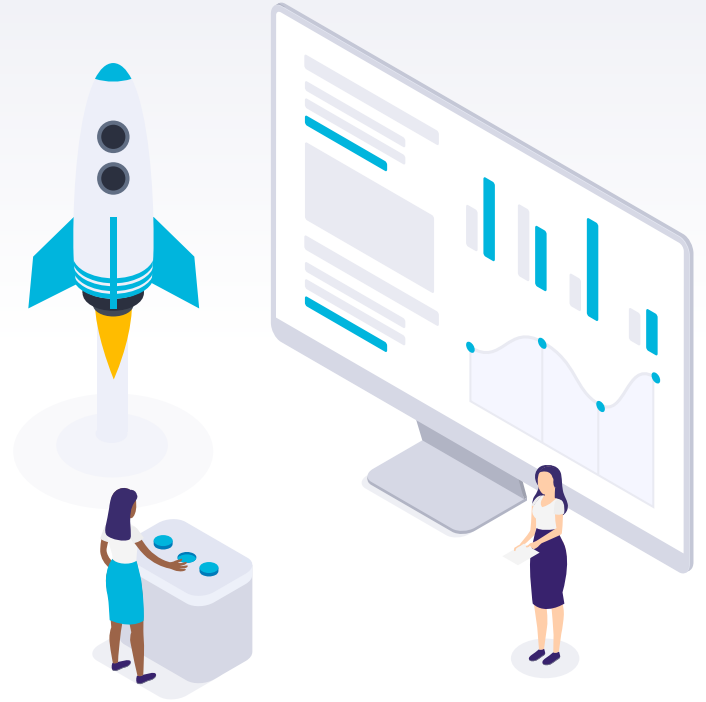
**Results**

- ▶ 7 participants deleted the analogy they created



4

# BUGS



# BUGS

- ▶ **Login Page**

Username form autofocuses, hiding registration button

- ▶ **Search Page**

“Back” button sometimes failed to save search query

- ▶ **Profile Page**

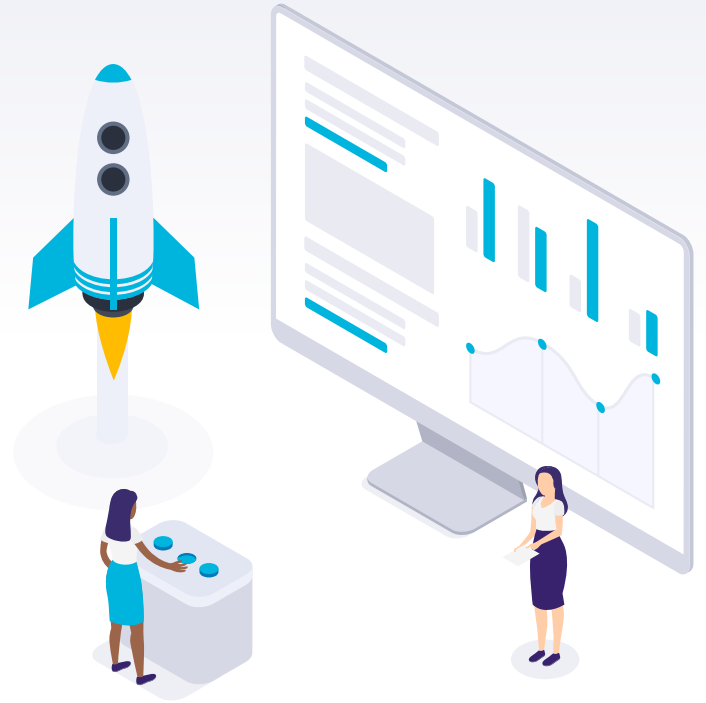
On analogy deletion, redirects to home page

**TOTAL BUGS** 3

**SEVERITY** Very Minor

5

# TOP RECOMMENDED CHANGES



# TOP RECOMMENDED CHANGES

- ▶ **“Favorite” Button**

“Bookmark” instead of “Heart”

- ▶ **Search Page**

Add UI “Return to Search” option

- ▶ **Create Analogy Page**

Automatically change tabs if user hasn’t input all fields

# THANKS!

Full report will be posted  
on my website!

