CS5760: Usability Test Report

Laura Albrant & Brandon Woolman

Table of Contents

Introduction

UI

This application is an attempt to create a more user-friendly introduction to regular expressions (regex). This is done by providing an adaptable canvas, on the right hand side, to assemble a block version of regex, including a block bank on the left hand side. Below this canvas and block bank is a field to show the generated regular expression and a table to input test text input. Additionally, there is a separate help page with brief description and short examples of how each block works.

Tests

For each participant, we provided ten separate trials to create a regular expression using the block language in the UI as well as test cases to verify what they created works to a degree. They were aware of a 3-minute time limit for each trial. They were also provided unlimited access to the help/tutorial page that described each block's function.

Test Goals

These goals are for all tests, regardless of scenario.

1. Determine how easy the interface is to understand - Progressively harder scenarios should provide a strong metric.
2. Determine which, if any, blocks cause confusion, or error, for participants.
3. Identify major issues with the prototype to be altered for the next iteration.

This goal is specific to the "Randomized" scenario (also known as Scenario 2).

1. This scenario has the additional goal to see whether or not the order of "progressively more difficult" has an impact on the user's learning experience.

Test Plans

Test Scenario 1

Description

This scenario consists of ten trials for the participant to complete. The trials will get more and more difficult. Each trial will consist of a vague instruction to create a simple or complex regular expression. Laura will be conducting this scenario for her 8 participants.

Participant Instructions

Thank you for coming in today. By being here, you are consenting to participate in our usability testing on a website involving regular expressions. You will be tasked with creating strings and testing regular expressions using a case matching function. When you are finished with each trial, clearly state “I’m done”. If you have questions during each trial, we ask that you first check the tutorial before asking us. You will have three minutes to complete each trial and will not be graded on accuracy. After we finish with the application, there will be a brief survey. At this time, I am happy to answer any questions you may have. If not, we should be finished with the testing within the hour.

Zoom will be used to record the screen as you complete the trials. Please open Zoom on your laptop and connect to the meeting with the ID: **357 239 7082**. Once that is done and you are ready, I will hand you the instructions for your first trial.

After 5 trials, do talk-aloud (instruction given purely verbally)
For the next few trials, we want you to tell us step-by-step what you are doing as you do it.

Trials

1. **Name:** Text Block
   Instruction:
   Use a Text Block to generate a regex string to capture the literal string of “Apple”.
   Match Case(s):

Text to Match	Should Match	Shouldn’t Match	Actual Result
Apple	☑	☐	✅

   Expected Solution:

2. **Name:** Start Block
   Instruction:
   Generate a regex to capture any string that starts with ‘a’.
   Match Case(s):

Text to Match	Should Match	Shouldn’t Match	Actual Result
ab	☑	☐	✅

   Expected Solution:

3. **Name:** End Block
   Instruction:
   Generate a regex to capture a string that ends with ‘b’.
   Match Case(s):

Text to Match	Should Match	Shouldn’t Match	Actual Result
ab	☑	☐	✅

   Expected Solution:

4. **Name:** Repeat Block
   Instruction:
   Generate a regex to capture a string with any number of the letter ‘a’.
   Match Case(s):

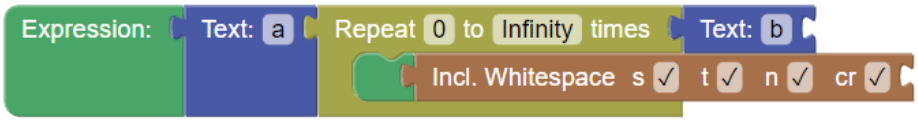
Text to Match	Should Match	Shouldn’t Match	Actual Result
aaaaaaa	☑	☐	✅

   Expected Solution:

5. **Name:** Include Whitespace
   Instruction:
   Generate a regex to capture a string that has any number of whitespace between the letter ‘a’ and the letter ‘b’.
   Match Case(s):

Text to Match	Should Match	Shouldn't Match	Actual Result
a b	☑	☐	✅

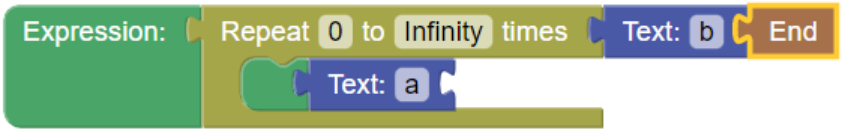
Expected Solution:



6. **Name:** Start + Repeat + End
Instruction:
Generate a regex to capture a string that starts any number of 'a' and ends with 'b'.
Match Case(s):

Text to Match	Should Match	Shouldn't Match	Actual Result
aaaaaab	☑	☐	✅

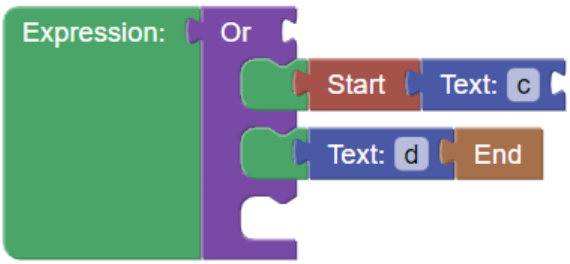
Expected Solution:



7. **Name:** Or - Start + End
Instruction:
Generate a regex to capture a string that starts with 'c' **or** ends with 'd'.
Match Case(s):

Text to Match	Should Match	Shouldn't Match	Actual Result
cab	☑	☐	✅
add	☑	☐	✅

Expected Solution:



8. **Name:** Repeat - Anything + Include Number
   Instruction:
   Generate a regex to capture strings that include a number somewhere after the first character.
   Match Case(s):

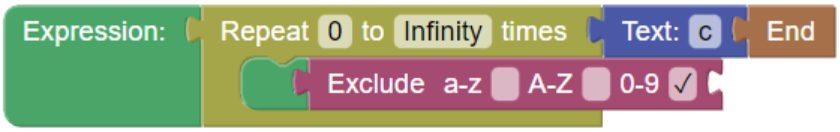
Text to Match	Should Match	Shouldn't Match	Actual Result
I have a 1 for you.	☑	☐	✅

   Expected Solution:

9. **Name:** Repeat - Exclude + End
   Instruction:
   Generate a regex to capture any string that ends with 'c' and contains no numbers.
   Match Case(s):

Text to Match	Should Match	Shouldn't Match	Actual Result
I do not have a number, c	☑	☐	✅
I have a 1 for you.	☐	☑	❌

   Expected Solution:



10. **Name:** Match a sentence
    Instruction:
    Generate a regex to capture any string that starts with "This", does not contain capital letters after "This", and ends in "1".
    Match Case(s):

Text to Match	Should Match	Shouldn't Match	Actual Result
This is the hardest 1	☑	☐	✅
This is NOT the hardest 1	☐	☑	❌
This is the hardest one	☐	☑	❌

Expected Solution:

Test Scenario 2

Scenario Description

This scenario consists of ten trials for the participant to complete. Each trial will consist of a vague instruction to create a simple or complex regular expression. Brandon will be conducting this scenario for his 8 participants.

Participant Instructions

Thank you for coming in today. By being here, you are consenting to participate in our usability testing on a website involving regular expressions. You will be tasked with creating strings and testing regular expressions using a case matching function. When you are finished with each trial, clearly state “I’m done”. If you have questions during each trial, we ask that you first check the tutorial before asking us. You will have three minutes to complete each trial and will not be graded on accuracy. After we finish with the application, there will be a brief survey. At this time, I am happy to answer any questions you may have. If not, we should be finished with the testing within the hour.

Zoom will be used to record the screen as you complete the trials. Please open Zoom on your laptop and connect to the meeting with the ID: **474 151 4369**. Once that is done and you are ready, I will hand you the instructions for your first trial.

After 5 trials, do talk-aloud(instruction given purely verbally)
For the next few trials, we want you to tell us step-by-step what you are doing as you do it.

Trials

The trials have the same content and instructions as Test Scenario 1 (see previous 2 pages). However, the order in which the trials are administered are randomized. This means that each participant who gets Test Scenario 2 may have a different order than the other participants who get Scenario 2.

Results

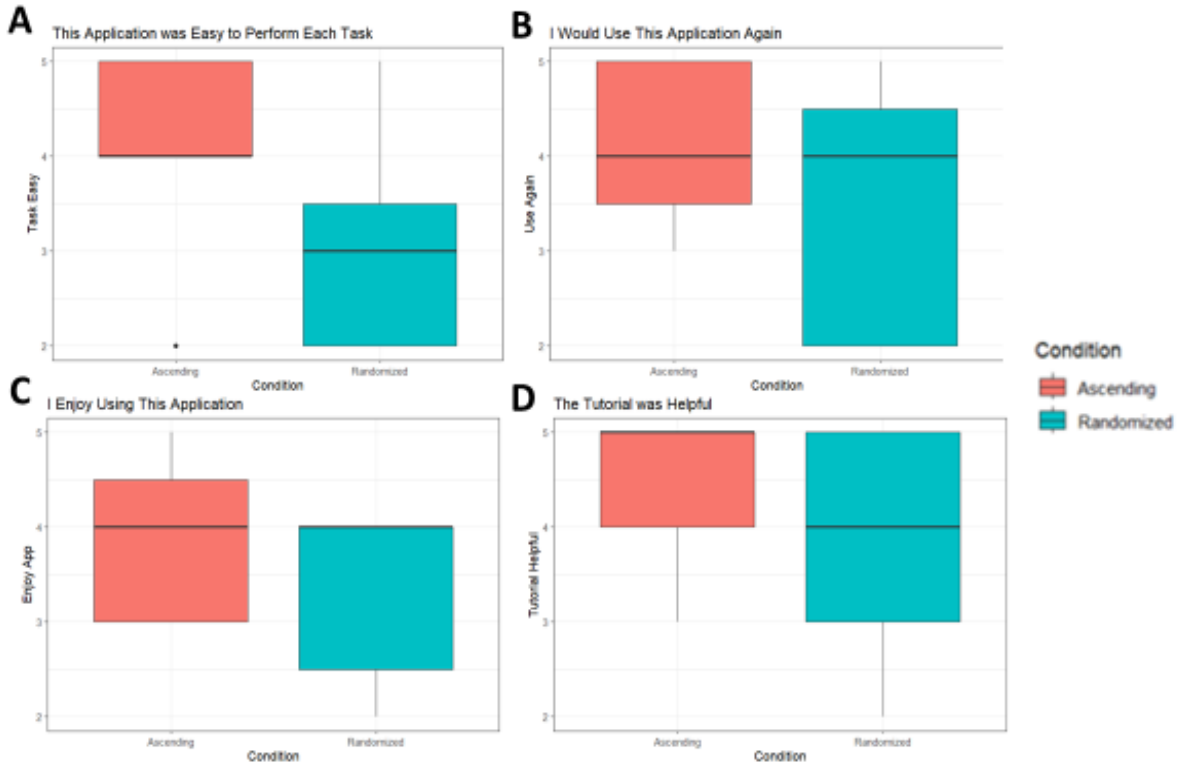
Participants

Recruitment was done through the computer science department at Michigan Technological University. Fourteen participants showed up to the testing. One participant failed to record their audio during the test and were excluded from the analyses. Participant majors ranged from engineering to computer science. Half of the participants included in analyses were graduate students.

Interview

On a likert scale of 1 (low) to 5 (high), the 14 participants rated readability a 4.5 which suggests that the application was very easy to read. The Regex familiarity was rated on a similar scale, however the average participant rated a 2.2 which means that most participants did not have much exposure to regular expressions. Below are the distribution of ratings, organized by scenario condition, for four other questions with similar likert scales.

Figure 1. Ratings for interview questions. Answers were rated on a likert scale of 1 (low) to 5 (high).



Scenario Results:

Figure 2. Tutorial Glances Across Conditions. This includes all completed trials.

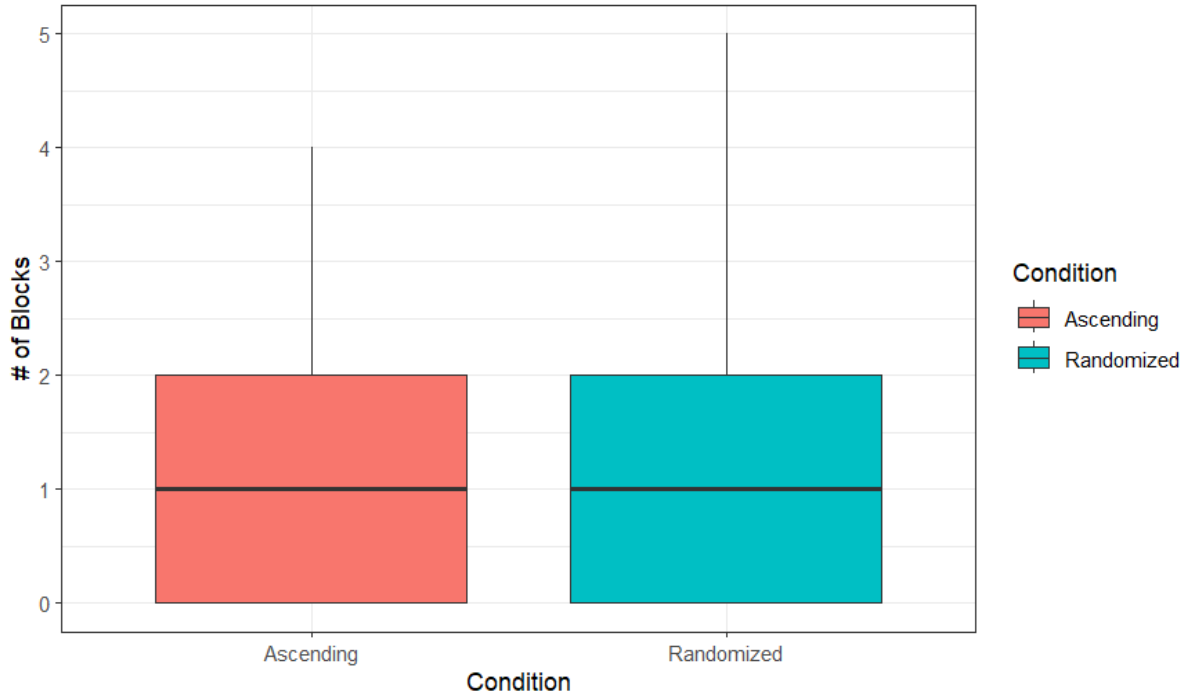


Figure 3. Unnecessary Blocks Used Across Conditions. This included all trials with correct answers. This was tested with an independent samples *t*-test and was found to be insignificant (*t*(51) = -1.50, p = .139).

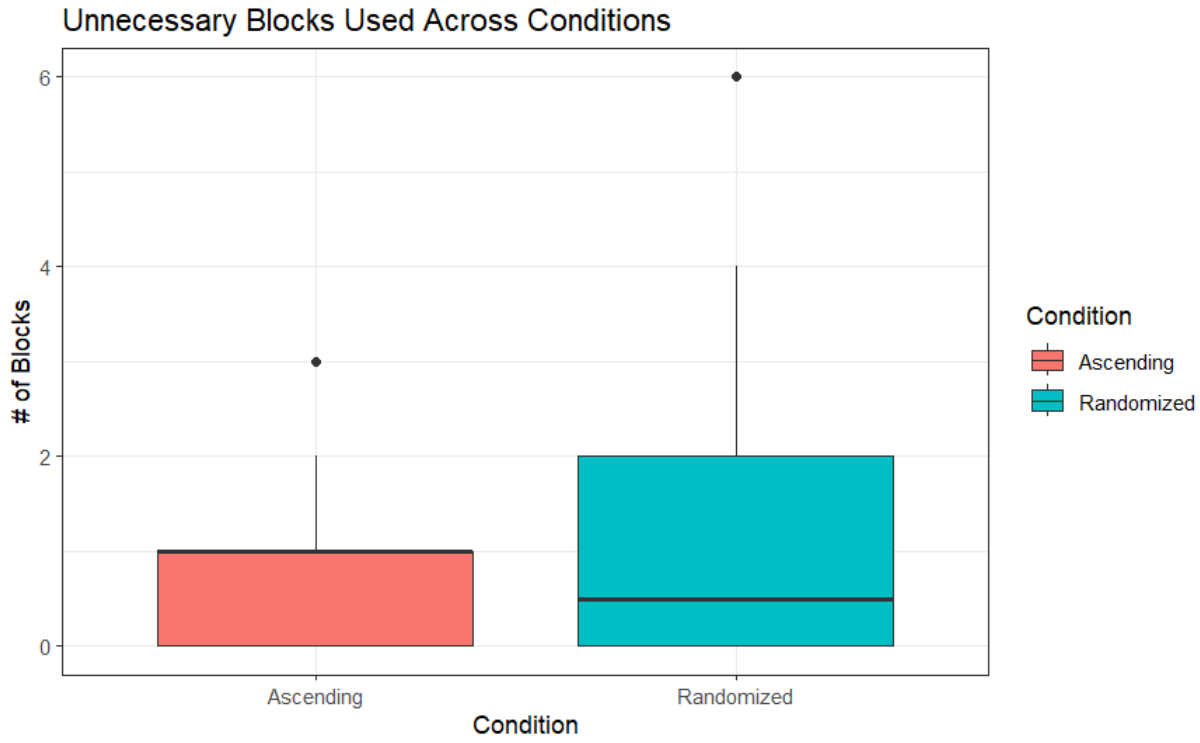


Figure 4. Number of Tutorial Glances Across Trials. This included all trials with correct answers. An independent samples *t*- test showed that performance across conditions was significantly different ($t(87) = 2.22$, $p < .05$).

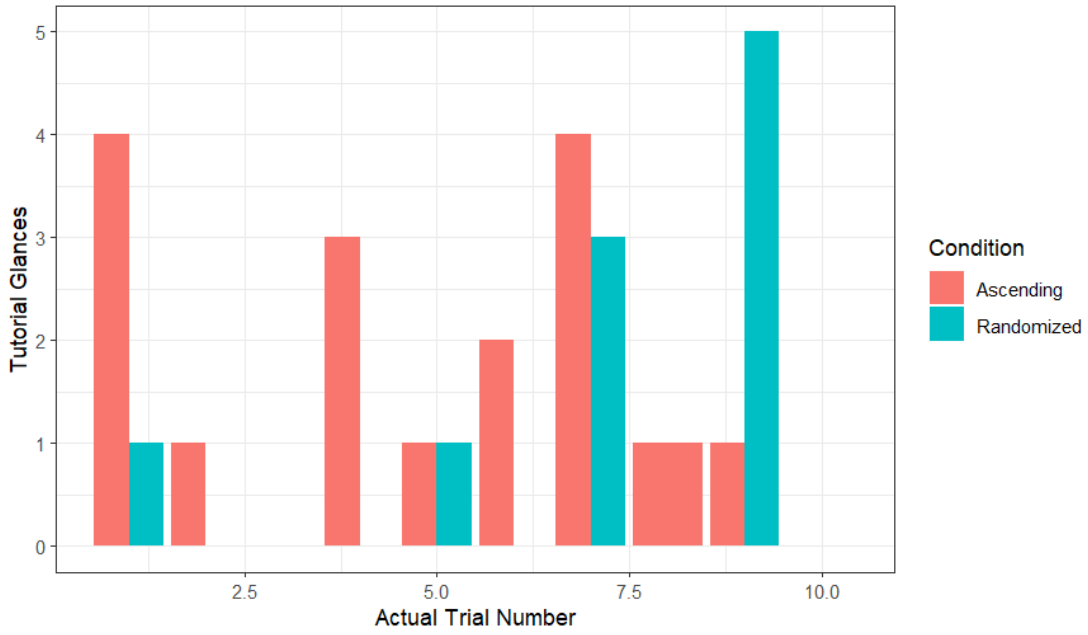


Figure 5. Time to Complete Trial. This includes all trials that participants stated that they were done. An independent samples *t*-test was used to test if there was a significant difference between condition ($t(78) = 2.35$, $p < .05$).

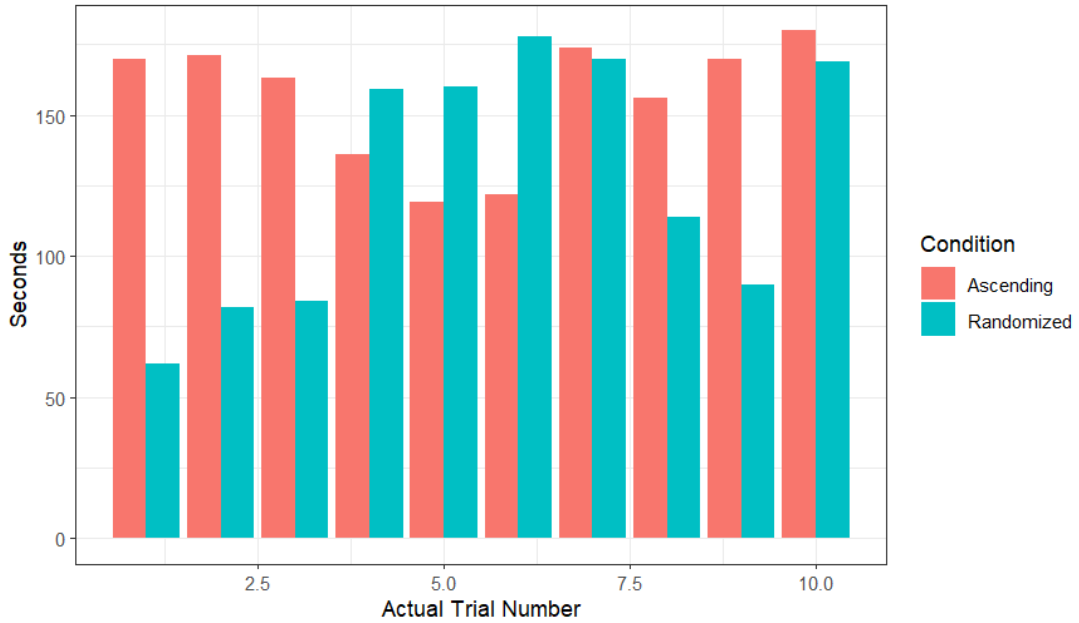
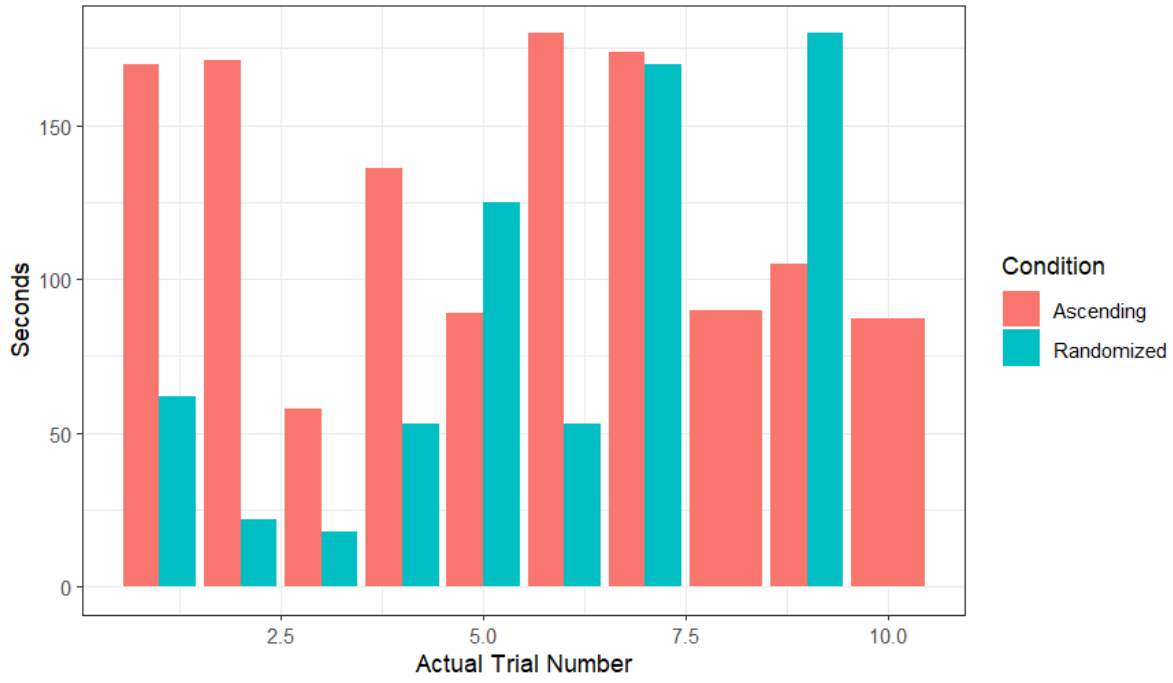


Figure 6. Time to Complete Trial If Scenario Correct. This builds off of the last graph, but instead of completed trials, we tested if the answers were correct. The difference between the groups were still significant ($t(44) = 2.45$, $p < .05$).



Think Aloud:

Block	Participant Confusion
Anything	• Uncertainty on if it applied to one character or more
Or	• Unsure as to why there was a third section to the block
Range Include/ Exclude	• Often treated as a repeat block • Misconception that the “range” refers to a range of characters/indexes from the input • Only one of the two inputs was filled in numerous times
Repeat	• Referred to as a “slider” by a participant • At least one participant tried to use it without any blocks inside it
(All)	• The existence of the tooltips is not obvious to most

Conclusions

The help page is a good starting point but needs numerous improvements. For starters, the help pages' block screenshots should actually match what is shown in the block bank. Numerous examples with multiple blocks and test case results should be included within the help page. The testing condition of ascending versus randomized revealed that these examples should be in ascending order by difficulty. Additionally, reorganizing the blocks, both in the bank and on the help page, could aid in clarifying how the blocks function to the users. This could look like creating tabs in the bank, where one of the tabs is for blocks that require nested blocks (i.e. repeat, or, etc.). Looking at the blocks individually, the *Range Include* and *Range Exclude* blocks were often misused, or not provided both inputs necessary. To remedy this, additional indication that both inputs are required to generate regex for that block should be given to the user.

As for the workspace/testing page, the original, autogenerated, *Expression* block should be able to be moved on its own. Currently, the entire canvas moves if the user attempts to move it. It could also be beneficial to lessen the time needed for the tooltips to popup for the blocks or to find a natural way to inform the user of their existence, as almost none of the participants became aware of their existence. Lastly, two separate 'clear' buttons should be added to the workspace. One of these buttons should clear everything. The other should only clear the test cases. Currently, the only way to reset anything is to manually delete blocks or refresh the screen.

Overall, we found that the application shows a lot of potential. There were multiple participants who were able to create correct regular expressions having not known regex beforehand. With the recommended improvements, we strongly believe that this application will achieve its purpose.

Appendices

A: Testing Attendance

Laura's Testing

Participant	Undergrad 1	Undergrad 2	Undergrad 3
1	Zane	Ame	
2	Zane	Ame	
3	Zane	Quentin	
4	Ame	Quentin	
5	Mya	Quentin	
6	Mya	Zane	
7	Mya	Zane	Maritza

Brandon's Testing

Participant	Undergrad 1	Undergrad 2	Undergrad 3
1	Isaac	Ame	
2	Isaac	Ame	
3	Ame	Noah	
4	Ame	Noah	
5	Maritza	Quentin	
6	Maritza	Quentin	
7	Isaac		

B: Bug Report

- The Beginning block is named Start in its screenshots on the help page.
- If you type in a test case and have no blocks beside Expression, the test case calls it a pass. It should, ideally, stay as the white background until there are blocks besides *Expression*.

C: Testing Challenges

- 1 Participant gazed at their emails in the middle of a trial
- 2 Participants opened 10+ tabs while switching between the app and help page
- 1 Participant had a faulty microphone which did not pick up voices while testing and in result was left out of analyses
- Generally there appeared to be a language barrier with a few of the participants which may have impacted the results
- The tested population was not the direct target of the application