STACK_OVERFLOW_AND_I: An Interactive Comparison Tool For Software Development Insights

Samantha Norrie

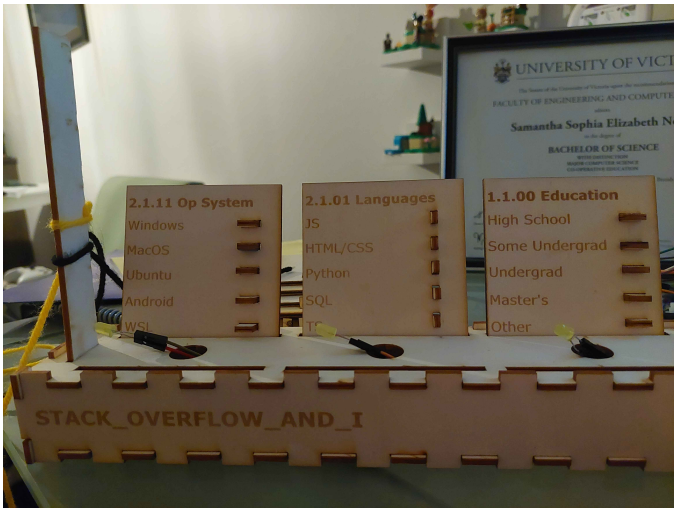

Fig. 1. STACK_OVERFLOW_AND_I's cardholder

Fig. 2. Input cards in the cardholder

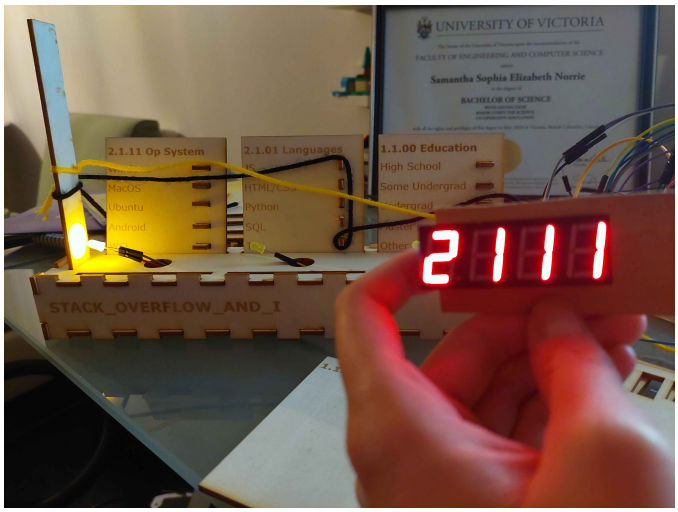

Fig. 3. Question 2.1.11 being selected in the Arduino system

I. Introduction and Motivation

With technology evolving rapidly, it is essential for professionals in the field to stay current with emerging trends, opportunities, and insights. Stack Overflow, a popular forum used by developers, surveys their users annually in order to support this [1], [2]. The 2024 edition of the survey had over sixty-thousand participants and covers topics such as popular technology for software development and the use of AI in the field. While the data from the survey can be viewed on Stack Overflow, it can be difficult to draw conclusions from the site itself due to the amount of information that is presented at once. STACK_OVERFLOW_AND_I aims to solve this issue by allowing users to physically query the existing dataset while also allowing them to insert their own data. This paper details the system, its uses, and how it can be reconstructed for other datasets.

II. Design and Workflow

STACK_OVERFLOW_AND_I contains three different interactive data components [3]:

1) Line charts for exploring personal data using questions from the survey (shown in Figure 3 and Figure 5)
2) Bar chart overlays for exploring data from the 2024 Developer survey (shown in Figure 5)
3) Joystick and display system for viewing details

Fig. 4. Answers woven into input cards

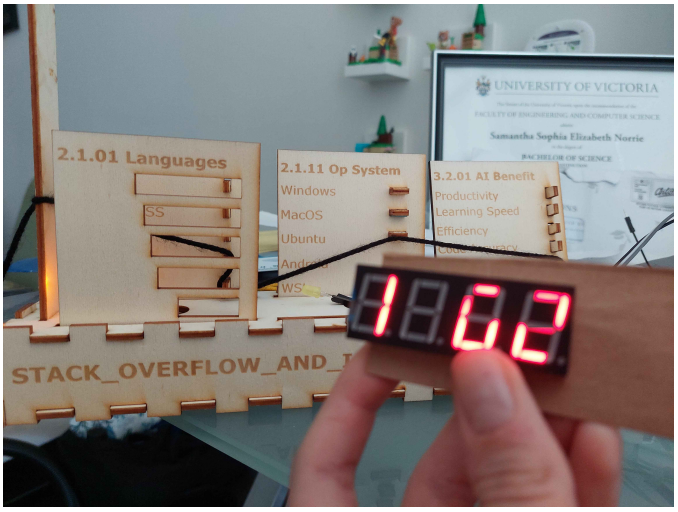

Fig. 5. The system's joystick and controller are used to see more detail on the data. The values appear blurry due to multiplexing, but they are visible in real life.

about data visualized in bar chart overlays (shown in Figure 4)

Component 1 and Component 2 use wooden cards representing the user's input and data (output) from the survey. The input cards each display a question asked in the survey. Each card has a title, which contains the number of the question in the survey and a shortened version of question itself. Underneath the title sits possible answers alongside small hooks. The users slot input cards into the back slots of the cardholder (seen in Figure 2), attach strings to the pole of the cardholder and weave the strings through the hooks in the input cards to 'select' their answers. If desired, the colours of the string used can be a visual primitive (seen in Figure 4). The output cards can then be slotted in front of the input cards

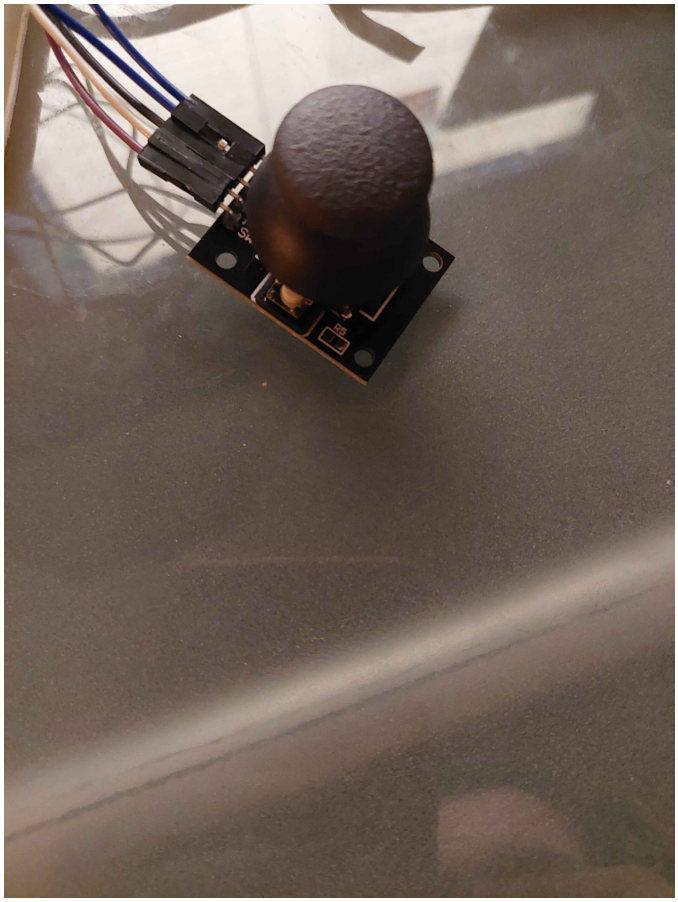

Fig. 6. The system's joystick

to show the results derived from the survey, allowing users to compare themselves against the Stack Overflow community

Some survey questions allow for users to select multiple answers. To make the selection of multiple answers easier, the input cards of multi-answer questions contain vertically placed hooks instead of horizontal hooks (seen in Figure 7).

Component 3, referred to as the Arduino system throughout this paper, allows for users to explore the data in more detail by allowing them to iterate through the percentage values for the labels of each question.

The following points detail how to use STACK_OVERFLOW_AND_I, assuming all components are used:

1) Select three questions from the 2024 Developer Survey and retrieve their respective input and output cards. The input cards have several hooks on them, and the output cards have rectangles cut out of them. The pair of input and output cards for each question can be identified by their matching

title.

2) Insert the input cards into the back slots of the cardholder (Figure 1 and Figure 2).
3) Enter questions into the STACK_OVERFLOW_AND_I system using the joystick (Figure 6) and display (Figure 3), with each question selected by its condensed code (e.g., 3.2.04 becomes 3204). Select the question you are selecting for by navigating left or right. The current question is denoted by the LED at its slot being lit (Figure 3).
4) Weave user answers using string attached to the cardholder pole (Figure 4).
5) Press the joystick to lock in the selected questions. Their percentage values will appear on the display.
6) Move the joystick left or right to view details of each question, indicated by its LED.
7) Move the joystick up or down to navigate through percentage values for the current question. The left side of the display shows the label index, and the right shows its percentage value (Figure 5).

To prevent the misreading of inputted data, it is recommended that users choose how they are going to weave their data into the input cards. The following system is recommended:

- When no answer is applicable, weave string behind the input card.
- For single-select questions and multi-select questions where only one answer is applicable, rest string on top of a label's hook to 'select' it.
- For multi-select questions where multiple answers are applicable, weave string on the right side of a label's hook to 'select', and weave string on the left side of a label's hook to not select it.

III. Construction

In addition to being useful for exploring data from the Stack Overflow Development survey, STACK_OVERFLOW_AND_I also presents a template that can be used for visualizing other datasets. The code, list of hardware components used, and laser cutting template for STACK_OVERFLOW_AND_I can be found on GitHub. The tool's construction is detailed in this section to help support this.

A. Cards and Cardholder

The cards and cardholder were created using laser-cut 3 mm plywood to ensure easy readability of the engraved text. Figure 8 shows the template used for STACK_OVERFLOW_AND_I. If a different material thickness were to be chosen, the rectangular slots used for holding the cards would need to be adjusted. An extra 0.1 mm was added to the slot width to allow easy card placement and removal, though this spacing may cause slight shifting; users may adjust by 0.0-0.1 mm based on preference.

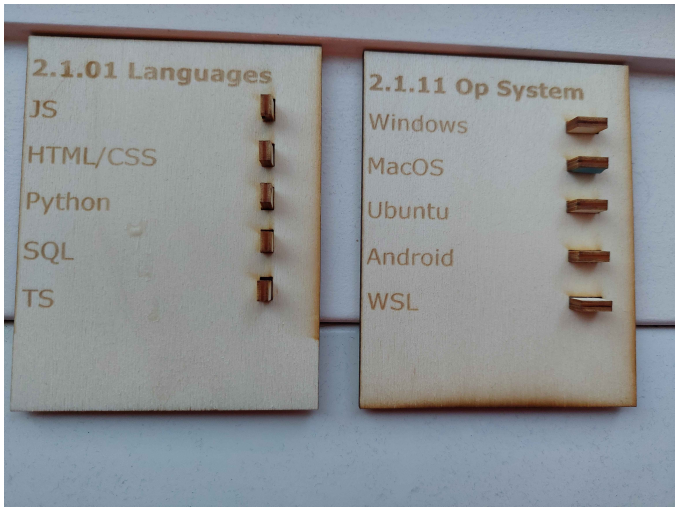

Fig. 7. Two of STACK_OVERFLOW_AND_I's input cards. The card on the left is designed to allow for multiple answers to be selected, while the card on the right is designed to only allow for one answer to be selected.

Users may also want to modify the cardholder's height and width to fit their electronics and card size. The template in Figure 8 accommodates electronics but requires a cardboard floor (Figure 9) to support cards at the current height.

Finally, it's recommended not to reduce the width between input and output card slots, as the current spacing allows input card labels to be visible behind the output cards.

B. Arduino system

The Arduino system enables users to view the percentage value of each label on the input cards. A challenge in development was using the joystick module with the display due to multiplexing. When multiplexing is used, multiple information channels are combined into one [4]. The display uses multiplexing for each of its LED components, and rapidly switches between them to create the illusion of multiple digits being shown simultaneously.

Because of this, the *delay()* function, typically used to prevent noise with the joystick, could not be used. A workaround is to implement debouncing manually with the *millis()* function and some arithmetic to limit how quickly inputs from the joystick are registered [5].

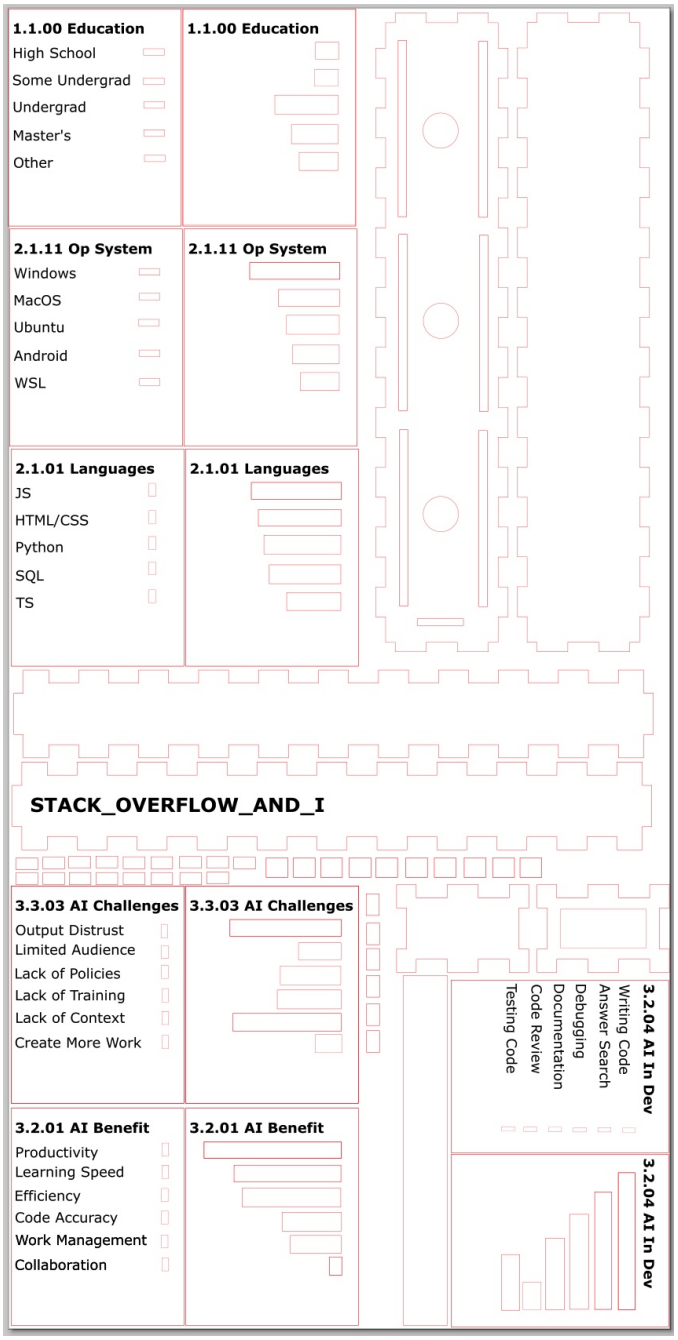

Fig. 8. The laser cutting template for STACK_OVERFLOW_AND_I.

IV. Limitations and Future Work

STACK_OVERFLOW_AND_I's fixed size limits the size of queries that can be done with the system. Future iterations of the system could include more slots, or allow for slots to be dynamically added with new hardware.

The strings used in the current prototype are thick, causing the input cards to become less legible as more strings are added. The hooks of the input cards also become overwhelmed when more than a few strings are hooked on them. This can be fixed by using thinner strings and/or increasing the depth of the hooks.

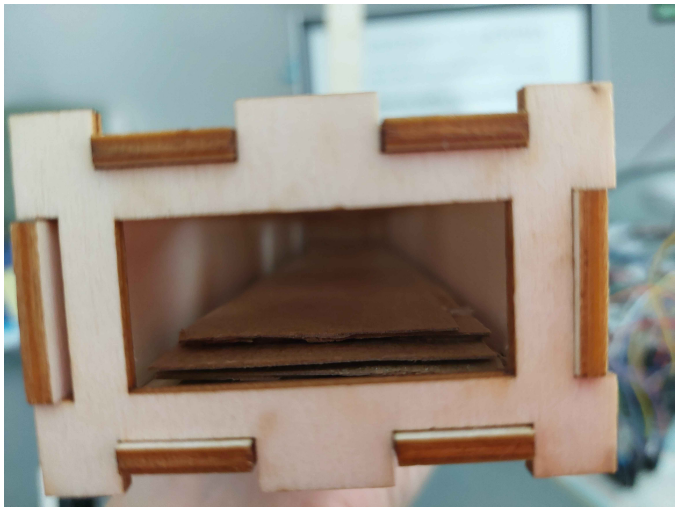

Fig. 9. The cardboard floor inside of the system

Due to the size of the input cards, questions with more than six labels had to be reduced down to having six or less labels. This was done by creating an 'other' category for single-select questions, and choosing the most popular/topical answers for multi-select questions. The data extracted from the survey and used in the current prototype can be seen in A.

Lastly, The percentages displayed in the display had to be rounded to two digits to allow for separation between the percentage values and their indices. The use of a bigger screen will allow for more digits to be used.

V. Related Work

The tool's design takes inspiration from multiple projects. Its line graph component takes inspiration from *Pulse*, an interactive line graph made of string [6]. The virtual combination of line charts and bar charts is inspired by the combinations seen in Microsoft's Excel, and in front-end libraries such as CanvasJS and ApexCharts [7]–[9].

The overall design and inspiration for the system came from the idea of multi-resolution. Multi-resolution is a technique used in Information Visualization to help prevent visual cluttering. Visual clutter refers to the use of unnecessary or an abundance of information in a visualization, which leads to the visualization being illegible or confusing [10]. Multi-resolution refers to the balancing of the amount of information that can be seen by a user at a given time.

VI. User Profiles

STACK_OVERFLOW_AND_I's target users are those who are involved in software development. The following subsections provide two sample user profiles.

A. Sample User Profile: A New Developer Looking for Insights

Nico, a self-taught software developer skilled in PHP, C, and SQL, is struggling to find a job that fits her skills and wonders if using AI might help her work. She has not used it yet as she fears doing so not being socially acceptable. To explore this, she uses STACK_OVERFLOW_AND_I, selecting input and output cards and examining survey data.

She quickly sees that many developers use languages like JavaScript and Python and recognize AI's benefits. This prompts her to start learning these languages and incorporating AI into her work.

B. Sample User Profile: A Project Manager Developing AI Policies

Usopp, a project manager, works with developers who use AI in their work, which has concerned his business team. They are worried about their software's quality worsening because of it, especially if it is used to test their software. To address this, Usopp uses STACK_OVERFLOW_AND_I to help him develop a policy that satisfies both teams. He selects AI-related questions from the card deck, adding input cards to the structure and marking developers' opinions with yellow string and the business team's with pink (Figure 4). He finds that developers don't use AI for testing but find it beneficial otherwise. The teams agree on a policy allowing AI use in development but banning it in testing

VII. CST Classification

STACK_OVERFLOW_AND_I is a CST that is both a data-physicalization tool as well as a tangible user interface (TUI). It is a CST because it positively helps users with varying experience levels draw new conclusions that can be either mini-c or little-c creative discoveries [11], [12]. It supports both solo and group creativity as the tool can be used alone or with a group [13]. VI-A demonstrates the tool being used for solo mini-c creativity as it allowed the user to have new realizations that are personally significant, and VI-B demonstrates the system being used for group little-c creativity because the user was able to come to realizations that helped his company solve a problem. It supports the *Look* stage of *Sawyer's Eight Stages of Creativity* because it allows users to draw new conclusions from the dataset.

STACK_OVERFLOW_AND_I's data physicalization qualification comes from it allowing for the exploration of physically encoded data in a way that helps users learn. It also meets the requirements of data-physicalization systems needing to be computer-supported as the Arduino system supports the numerical exploration of the encoded data.

Lastly, STACK_OVERFLOW_AND_I also qualifies as a TUI because it allows users to physically input their own data in a way that is constructive, spatial, and relational [14].

References

[1] Stack Overflow, "Stack overflow annual developer survey." [Online]. Available: survey.stackoverflow.com

[2] ——, "2024 developer survey." [Online]. Available: survey.stackoverflow.co/2024/developer-profile/

[3] W. Peng, M. Ward, and E. Rundensteiner, "Clutter reduction in multi-dimensional data visualization using dimension reordering," *IEEE Symposium on Information Visualization (INFOVIS)*, 2004.

[4] U.S. Department of Federal Highway Administration, "Chapter 2. fundamentals of telecommunications." [Online]. Available: ops.fhwa.dot.gov/publications/telecomm_handbook/chapter2_03.html

[5] J. Charles, "What is debouncing?" [Online]. Available: medium.com/@jamischarles/what-is-debouncing-2505c0648ff1

[6] C. Ferrara and J. McTaggert, "Pulse: Tangible line graph," 2012.

[7] The Microsoft 365 Marketing Team, "Combining chart types, adding a second axis," 2012. [Online]. Available: www.microsoft.com/en-us/microsoft-365/blog/2012/06/21/combining-chart-types-adding-a-second-axis/

[8] canvasJS, "Combination of javascript line, area and column charts," 2024. [Online]. Available: canvasjs.com/javascript-charts/column-line-area-chart/

[9] ApexCharts, "Line column," 2024. [Online]. Available: apexcharts.com/react-chart-demos/mixed-charts/line-column/

[10] R. Rosenholtz, Y. Li, and L. Nakano, "Measuring visual clutter," *Journal of Vision*, 2007.

[11] S. Somanath, "Designing creativity support tools," 2024.

[12] M. Grohman, "Teaching for creativity: Mini-c, little-c and experiential learning in college classroom," 2018.

[13] E. F. Rietzschel and B. A. Nijstad, "Group creativity," *Encyclopedia of Creativity*, 2020.

[14] B. Ullmer and H. Ishii, "Emerging frameworks for tangible user interfaces," *IBM Systems Journal*, 2000.

Appendix

	Question 1	Question 2	Question 3	Question 4	Question 5	Question 6
Question Code	1.1.00	2.1.01	2.1.11	3.2.01	3.2.04	3.3.03
Question	Which of the following best describes the highest level of formal education that you've completed?	Which programming, scripting, and markup languages have you done extensive development work in over the past year?	What is the primary operating system in which you work?	For the AI tools you use as part of your development workflow, what are the MOST important benefits you are hoping to achieve?	Which parts of your development workflow are you currently using AI tools for?	What are the challenges to your company/whole team using AI code assistants or GenAI tools?
Category	Developer Profile	Technology	Technology	AI	AI	AI
Single-Select (Y/N)	Y	N	N	N	N	N
Label 1	High School	JS	Windows	Productivity	Writing Code	Output Distrust
Label 1 Value	10%	62%	60%	81%	82%	66%
Label 2	Some Undergrad	HTML/CSS	MacOS	Learning Speed	Answer Search	Limited Audience
Label 2 Value	13%	53%	32%	62%	68%	26%
Label 3	Undergrad	Python	Ubuntu	Efficiency	Debugging	Lack of Policies
Label 3 Value	41%	51%	28%	59%	57%	32%
Label 4	Master's	SQL	Android	Code Accuracy	Documentation	Lack of Training
Label 4 Value	26%	51%	18%	30%	40%	31%
Label 5	Other	TS	WSL	Work Management	Code Review	Lack of Context
Label 5 Value	10%	39%	17%	25%	13%	63%
Label 6	NA	NA	NA	Collaboration	Testing Code	Create More Work
Label 6 Value	NA	NA	NA	8%	27%	13%

Fig. 10. The extracted data from the 2024 Developer Survey used in the STACK_OVERFLOW_AND_I prototype.