Usability of Error Messages for Introductory Students

Paul A. Schliep
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
schli202@morris.umn.edu

ABSTRACT

Error messages are an important tool for programmers to help find and fix mistakes or issues in their code. When an error message is unhelpful, it can be difficult to find the issue and may impose learning difficulties. Error messages are especially critical for introductory programmers in understanding problems with their code. Not all error messages are beneficial for helping novice programmers, however. This paper discusses the general usability of error messages for introductory programmers, analyses of error messages in compilers and DrRacket, and two methodologies meant to improve error handling.

Keywords

Novice programmers, usability, error messages, usability studies, compiler errors, syntax errors

1. KEY POINTS

The question being addressed in this research is to figure out how well error messages in programming languages help introductory students find mistakes in their programs. This question is being addressed through various methods of evaluating the usability of the error messages and the methodologies in which they can be or potentially have been improved. The methods of evaluating the error messages include usability studies, trials, and developed rubric to assess them, and some of the methodologies that suggested improvements to the messages include a set of suggested principles for compiler errors, enhancement of syntax errors, and proposed recommendations for exception messages.

The sources for this paper each propose or analyze something new to help address this problem while still being related. On Compiler Error Messages: What They Say and What They Mean discuss the current state of compiler error messages, how that affects novice programmers, and what they suggest to resolve those issues. Enhancing Syntax Error Messages Appears Ineffectual talks about the implications of syntax error messages are on how introductory CS students can fix errors in their programs and they present an enhanced library of compiler messages specifically for syntax errors and analyze its effectiveness. Mind Your Language: On Novices' Interactions with Error Messages and Measuring the Effectiveness of Error Messages Designed for Novice Programmers both evaluate error messages in the Racket programming language and discuss how well they help introductory students fix mistakes in their programs, and how to apply their developed rubric to improve these error messages. Each paper discusses an analysis of different types of error messages and in what ways they may be resolved, and by studying these analyses and their respective methodologies, we can be able to compare and contrast their methods and results.

Error message usability has been tested for many years, but it has gained more attraction in recent years with the rising studies in HCI. Through these studies, developers and professors have been able to implement error message libraries and test them, but not always with conclusive results as seen in my source, Enhancing Syntax Error Messages Appears Ineffectual where the number of students' syntax errors did not fall.

In order to explain some of the analyses of the error messages, the reader will need to be informed of concepts such as usability studies and compiler errors.

2. REFERENCES

[1] P. Denny, A. Luxton-Reilly, and D. Carpenter. Enhancing syntax error messages appears ineffectual. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, pages 273–278, New York, NY, USA, 2014. ACM.

[2] P. Denny, A. Luxton-Reilly, E. Tempero, and J. Hendrickx. Understanding the syntax barrier for novices. In *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, ITiCSE '11, pages 208–212, New York, NY, USA, 2011. ACM.

[3] B. Hartmann, D. MacDougall, J. Brandt, and S. R. Klemmer. What would other programmers do: Suggesting solutions to error messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1019–1028, New York, NY, USA, 2010. ACM.

[4] B. S. Isa, J. M. Boyle, A. S. Neal, and R. M. Simons. A methodology for objectively evaluating error messages. In *Proceedings of the SIGCHI Conference on Human*

UMM CSci Senior Seminar Conference, December 2013 Morris, MN.

Factors in Computing Systems, CHI '83, pages 68–71, New York, NY, USA, 1983. ACM.

[5] S. K. Kummerfeld and J. Kay. The neglected battle fields of syntax errors. In *Proceedings of the Fifth Australasian Conference on Computing Education - Volume 20*, ACE '03, pages 105–111, Darlinghurst, Australia, Australia, 2003. Australian Computer Society, Inc.

[6] G. Marceau, K. Fisler, and S. Krishnamurthi. Measuring the effectiveness of error messages designed for novice programmers. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education*, SIGCSE '11, pages 499–504, New York, NY, USA, 2011. ACM.

[7] G. Marceau, K. Fisler, and S. Krishnamurthi. Mind your language: On novices' interactions with error messages. In *Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Onward! 2011, pages 3–18, New York, NY, USA, 2011. ACM.

[8] C. Murphy, E. Kim, G. Kaiser, and A. Cannon. Backstop: A tool for debugging runtime errors. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '08, pages 173–177, New York, NY, USA, 2008. ACM.

[9] V. J. Traver. On compiler error messages: What they say and what they mean. In *Advances in Human-Computer Interaction*, 2010.

[1] [3] [4] [5] [6] [7] [8] [9] [2]