

Human-Computer Interaction through Code

by Jason T. Jacques <jtj2@st-and.ac.uk>

Background

Integrated development environments (IDEs) are much the same: a large editor window in the centre, code browser on the left, maybe a pallet on the right, some output down below, and a few toolbars at the top. No matter the language or platform, they largely follow the same pattern. While previous research has suggested large scale changes to this paradigm (DeLine and Rowan, 2010), these overhauls have seen limited acceptance.

Familiarity and maintaining a transferable skill-set almost certainly have some part to play in this rigid environment. These considerations have limited the scope for new human-computer interaction techniques and guidelines to be applied to professional development tools.

Text presentation in IDEs recently became a topic of interest among the developer community when Microsoft's most recent Visual Studio offerings took a measured step away from established practice, to much critique, choosing to render much the interface text in block capitals. This interest highlights the significant impact of even subtle changes to the developer environment.

Research Question & Aim

The purpose of this research is to identify key features of the textual presentation of computer code that have a measurable impact on developer performance. The aim is to use these features to inform the development of a subtle, assistive code editor to improve the experience of programmers.

Method

A number of focused, controlled studies will be carried out to discover subtle interaction cues and code augmentation techniques that effect an improvement in developer performance (errors made, time taken, efficiency of solution, etc.). These results will be used to develop an augmented development environment, to be tested with developers, to ensure the external validity of the findings in a more typical development environment.

Participants in the initial studies will be gathered using crowdsourcing systems to allow large number of developers to experience the changes in a synthetic, but highly controlled environment. These initial studies will determine which

features are significant for the latter more comprehensive prototyped system and will provide the necessary background to develop a suitable experimental environment.

Impact

The increasingly pervasive nature of digital and computerised technology in our everyday lives has brought a new importance to the ability and efficiency of those who are responsible for the software that runs our world. Improving programmer performance, reducing bugs and increasing output, is vital to maintain a competitive workforce who are able to maximising the benefits of technology (Davies, 2005).

Understanding how developers perceive and interpret code is essential to improving the digital environment in which they work. The unexplored subtle interaction cues, which are the focus of this proposal, have the potential to non-invasively improve the experience of professional programmers and aid in the training of the next generation of software developers.

References

- Davies, W., 2005. Modernising with Purpose: A manifesto for a digital Britain. Institute for Public Policy Research.
- DeLine, R., Rowan, K., 2010. Code canvas: zooming towards better development environments, in: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 2, ICSE '10. ACM, New York, NY, USA, pp. 207–210.