MaliciousIDE – software development environment that evokes emotions

Michal R. Wrobel
Faculty Of Electronics, Telecommunications
And Informatics, Gdansk University of
Technology, Poland
Email: wrobel@eti.pg.edu.pl

Adam W. Zielke
Faculty Of Electronics, Telecommunications
And Informatics, Gdansk University of
Technology, Poland
Email: adam.wojciech.zielke@gmail.com

Abstract—Emotions affect every aspect of human live, including work. Numerous studies in software engineering have shown that negative emotions can lower the productivity of programmers. Unlike traditional approaches to managing software development, modern methods, such as Agile and Lean, take into account human aspects of programming. To thoroughly investigate the impact of negative emotions on the work of programmers, a malicious integrated development environment (IDE) was developed. This tool allow a observer to trigger malicious behavior of the IDE. Conducted study have proved its usefulness. Participants reported that it mostly invoked frustration and angry.

I. Introduction

RESEARCH on the role of emotions in the work of software developers has been deeply conducted in recent years. Finding the relationship between emotional state and productivity, or the quality of the developed code, may lead to the new methods of IT project management, and consequently to improve software development processes. This type of research corresponds to the current trend of research on social aspects in software development projects [1].

The field of affective computing in recent years has provided many algorithms and tools for recognizing the emotions of computer users. However, despite continuous development, they are still imperfect, especially in applications in unusual situations. Therefore, there is a huge demand for input channel data, such as a video clips or biometric data that are labelled by emotions. They can be used to teach algorithms or check new methods of recognizing emotions. Such data sets, especially those consisting of multiple channels, are not widely available to the public.

In the case of the analysis of emotions in the work of programmers, generic solutions are mainly used. With popular algorithms, such as recognizing emotions based on facial expressions, this approach is acceptable. However, the specificity of the work of programmers allows the development of other, dedicated algorithms. One of such approaches is recognizing emotions based on the analysis of patterns of typing and mouse movements [2]. To develop and learn such algorithms, it is necessary to gather data labelled with emotions. However, such data are extremely difficult to obtain from the real working environment. Therefore, laboratory studies are used for this purpose.

Quigley et al. identified thirteen laboratory emotions induction techniques [3]. However, most of them are not suitable for evoking the emotions of programmers during work. They usually rely on inducing emotions through exposure of the participant to visual or audible stimulus. The exception are three approaches: *peripheral physiological manipulations*, *motivated performance* and *physically real stimuli* [3]. In the case of software engineering studies, the first one is not suitable for general use because it requires direct stimuli in physiological systems, e.g. by injecting drugs. Others can be adapted to induce emotions of programmers.

For the purposes of laboratory studies on the emotions of programmers, a malicious integrated development environment, *MaliciousIDE*, has been developed at the Gdańsk Technical University since 2015. The goal of the project is to provide a solution that will induce negative emotions of programmers while working. It is adaptation of the *physically real stimuli* induction technique. In addition, the *motivated performance* was studied during a validation, which involved putting participants under the time pressure.

The rest of the paper is organized as follows: Section II describes the methods used so far in recognizing the emotions of software developers, Section III describes the concept of the MaliciousIDE plug-in, a survey that verified assumptions and architecture of the solution. In Section IV a study in which MaliciousIDE was used as a method to evoke emotions is described. Finally, Section V concludes.

II. Background

So far, a number of research has been conducted in the field of software engineering, which involved recognizing the emotions of IT teams members. Numerous attempts were made to identify emotions using various available channels. However, to the best of our knowledge, there have not yet been proposed methods of inducing emotions developed for the specifics of the work of programmers.

In some studies, general approaches to evoking emotions were used. Khan et al. in their studies tried to induce emotions of software developers by showing video clips, before performing the tasks [4][5].

However, in most studies there were no attempts to evoke emotions at all. In most cases, the researchers only collected

the data useful for recognizing emotions and, at most, obtained information on emotional states based on self-assessment of participants.

The most comprehensive research on utilizing physiological sensors during software developers work was conducted by Müller and Fritz[6][7][8]. During their study [6] on 17 software developers, they collected the following data: electroencephalography (EEG) using a Neurosky MindBand sensor, temperature, electrodermal activity (EDA) and blood volume pulse (BVP) using an Empatica E3 wrist band, and eye-tracking using Eye Tribe. The results of the experiment showed that EDA tonic signal, the temperature, brainwave frequency bands, and the pupil size were most predictive to classify progress of software developers, and brainwave frequency bands, the pupil size, and the heart rate to classify their emotions. Nevertheless, they noted strong individual differences with respect to the correlation and classification of physiological data [6]. Similar differences have also been found in our other studies on the use of sensors to monitor the physiology of computer game players [9].

A relatively new approach, which can be well suited to recognize the emotions of programmers is keystroke dynamics and mouse movements analysis. It is completely non-intrusive and does not require any additional hardware [10]. There have already been attempts to use this method to monitor software developers [2].

However, one of the most popular methods of recognizing emotions, widely used also in the field of software engineering, is the analysis of facial expressions [11][12][13]. It has gained popularity mainly as a universal and non invasive approach. Algorithms analyze video frames to identify face muscle movements and based on the Facial Action Coding System (FACS) [14] assess user's emotional state.
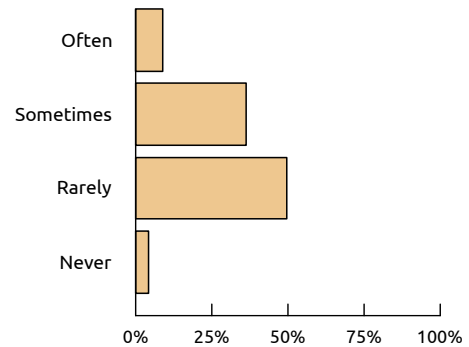
III. SOLUTION

In studies that used real world stimuli methods to evoke emotions, researchers used, among others, widely regarded as horrible, or disgusting animals (spiders, snakes), extreme sports (sky diving, mountaineering), or various types of food [3]. In order to exert an equally significant impact on the emotional state of software developers, we decided to prepare a programmer's nightmare – an unstable programming environment.

An integrated development environment (IDE) is an software which is used to write code, build project and run it. It is a fundamental tool used by the vast majority of programmers. To verify the importance of IDE for the work of programmers and check if its behaviour can affect emotional states, a survey was carried out on a group of 44 software developers, most of whom (61.4%) have professional experience longer than 4 years. Respondents answered three questions regarding the impact of the programming environment on emotions:
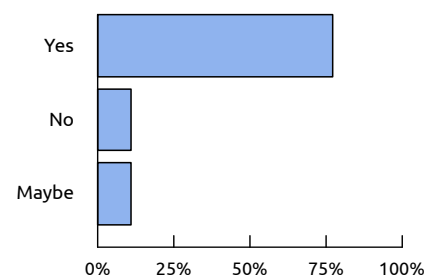
- How often do you encounter errors in the IDE?
- Would you change the IDE if it hinders your work?
- Are you getting angry when you can not complete the task because of problems with IDE?

Provided answers, shown in Fig. 1, confirmed that the IDE work and behaviour has an impact on the emotions of programmers. Among all, 68.2% of respondents confirmed that problems with IDE make them angry, and 77.3% would consider changing the IDE in such a situation.
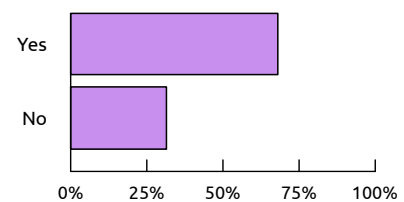

Fig. 1. Survey on the impact of IDE on the emotions of programmers.

In addition, information on the errors encountered by respondents in development environments were collected. It allowed to define seven malices that has been implemented in MaliciousIDE:

1) Clean the clipboard.
2) Delete all semicolons.
3) Freeze whole environment.
4) Hide IDE window.
5) Duplicate keys pressed by participant
6) Minimalize IDE window.
7) Move mouse pointer.

Instead of developing MaliciousIDE from scratch, it was decided to use the plug-in mechanisms, which allows third-party developers to extend the functionality of the program.

The NetBeans was chosen as IDE, on the basis of which malicious behaviour was implemented in the form of a plug-in. It is Open Source, well recognized environment written in Java language.

Preliminary tests of the tool have shown that automatically triggered malices may cause insufficient number of events. For example, users may not notice that the content of the clipboard have been cleaned if it was not used in a particular task. Therefore, it was decided that malices will be triggered manually by the observer, using a remote dashboard (Fig. 2). Thus, an appropriate number of events was ensured. Too few events might not induce emotions, but too much could lead to the disclosure of the malicious activity of the observer.
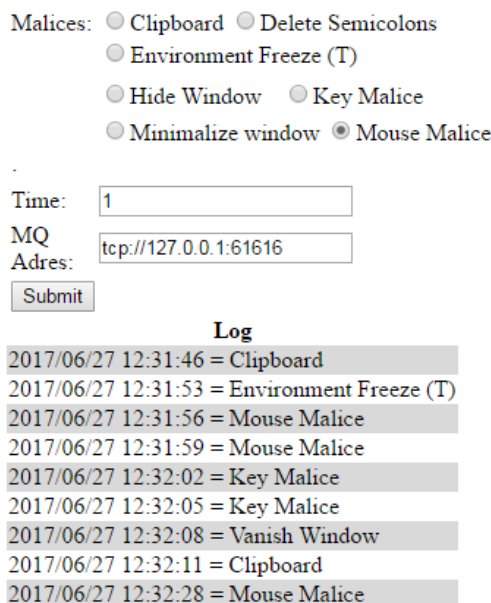

Fig. 2. MaliciousIDE dashboard

Both the plug-in and the dashboard were developed in Java, and in the latter, the Struts 2 framework was additionally used. In order to ensure reliable communication between the two parts of the tool, the Apache ActiveMQ message broker was utilized. MaliciousIDE is released as an Open Source project[1] and can be used freely to conduct research on the emotions of software developers.

IV. VALIDATION

The developed tool, MaliciousIDE, was used during the study conducted in April and May 2017 at a biometric stand [15] in the "Laboratory of Innovative IT Applications" at Gdansk University of Technology (GUT). The aim of the study was to check which input channels can be used to recognize the emotions of programmers during work [16]. Together 35 undergraduate computer science students, including 6 women
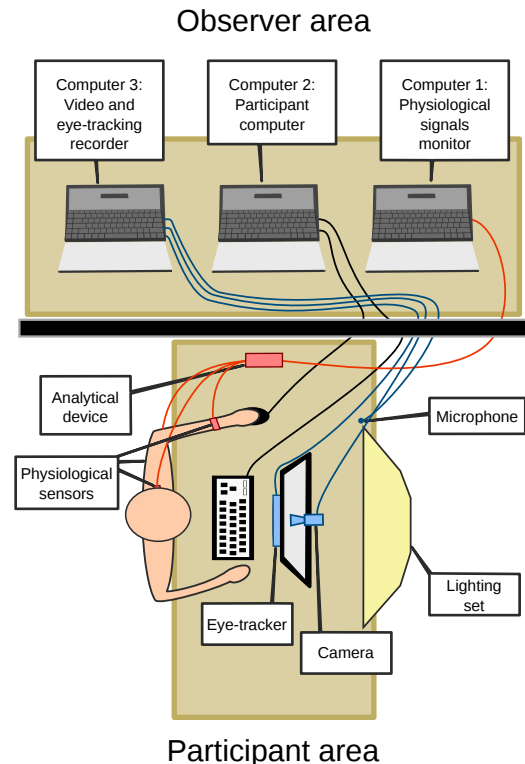
[1]http://www.popi.pl/maliciouside

Fig. 3. Study stand

and 29 men, participated in the study. A single session consisted of 5 programming tasks and lasted between 30 and 45 minutes.

During the entire session, the observer was present and controlled MaliciousIDE to evoke the participants' emotions. The most commonly triggered malices were: adding additional characters while entering text, changing the position of the mouse pointer, freezing the environment for 7 seconds, clearing the contents of the clipboard, and temporarily hiding the IDE screen. These actions were carried out to disrupt the work, but in a way that would seem to be a natural behaviour of the application. The frequency of events was manually adjusted so that users remain unaware of the observer's intended actions.

In addition, during the last task of the session, the observer was verbally announcing that the time required to complete the task must be shortened. The purpose of this action was to induce time pressure.

The room was divided into two parts, one for the participant and one for the observer (Fig. 3). On the participant's desk there was a monitor, a keyboard and a mouse connected to the Computer 2, on which the tasks were performed. At the top of the monitor a video camera was located, followed by a lighting set, supplied with Noldus FaceReader software, which was used to recognize emotions based on facial expressions [17]. Underneath the monitor eye-tracker device was located. A number of sensors were attached to the participant body, and were linked through the Coder FlexComp Infiniti by

Thought Technology analytical device with the Computer 1, which was located in the observer area. The observer also has a monitor, a mouse, and a keyboard connected to the participant's Computer 2. A data acquisition program for keystrokes' analysis was also available on the same computer [2]. MaliciousIDE Dashboard was running on Computer 3 and was used to control NetBeans plug-in on participant computer. It also collected all other data for the emotions recognition.

After completing all tasks, the participant was asked to complete a survey implemented using the Google Forms service. Among other questions, participants were asked to name the emotions caused by the unstable IDE. In the answers the most frequently appeared irritation (42.86%), anger (28.57%), followed by nervousness (25.71%) and frustration (11.43%). Four of the respondents (11.43%) indicated amusement (Fig. 4). A post-study informal interview revealed that it was related to the fact that these participants figured out that this unstable work was due to the deliberate action of the observer. The other emotions were pointed out by only one or two people.

On the other hand, the attempt to simulate the approaching deadline by shortening the time of the last task almost completely failed. Nearly half of the respondents indicated that this had no effect on their emotions at all. However, 7 participants indicated that shortening the time was a mobilizing factor, thus having a positive impact on the work attitude.
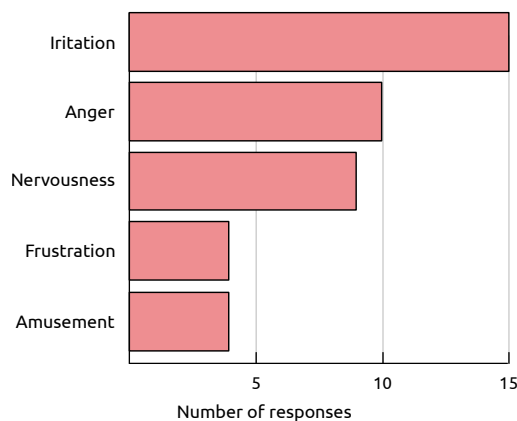

Fig. 4. Emotions induced by malicious behavior of the IDE

V. Summary

MaliciousIDE has been developed to allow researchers to induce the emotions of programmers in a laboratory environment. The presented approach and developed tool can be included to the group of real world stimuli methods of inducing emotions, as it provides a nightmare of programmers – an unstable development environment, which is additionally controlled by the observer.

During the study, emotion inducing method based on the MaliciousIDE has been evaluated. The developed plug-in has proven to be successful mechanism to trigger negative emotions of programmers. While the second approach, shortening the duration of one task, did not turn out to evoke emotions on a significant number of participants.

References

[1] A. Przybyłek and W. Kowalski, "Utilizing online collaborative games to facilitate agile software development," in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS'18).* IEEE, 2018.

[2] A. Kolakowska, "Towards detecting programmers' stress on the basis of keystroke dynamics," in *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on.* IEEE, 2016. doi: 10.15439/2016f263 pp. 1621–1626.

[3] K. S. Quigley, K. A. Lindquist, and L. F. Barrett, "Inducing and measuring emotion and affect," *Handbook of Research Methods in Social and Personality Psychology*, p. 220–252, 2014. doi: 10.1017/CBO9780511996481.014

[4] I. A. Khan, "Mood Independent Programming," in *ECCE '07. Proceedings of the 14th European conference on Cognitive ergonomics: invent! explore!*, no. August, 2007. doi: 10.1145/1362550.1362606 pp. 269–272.

[5] I. A. Khan, W.-P. Brinkman, and R. M. Hierons, "Do moods affect programmers' debug performance?" *Cognition, Technology & Work*, vol. 13, no. 4, pp. 245–258, 2011. doi: 10.1007/s10111-010-0164-1

[6] S. C. Müller and T. Fritz, "Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress," in *Software Engineering (ICSE), 2015 IEEE/ACM 37th IEEE International Conference on*, vol. 1. IEEE, 2015. doi: 10.1109/icse.2015.334 pp. 688–699.

[7] T. Fritz and S. C. Müller, "Leveraging biometric data to boost software developer productivity," in *Software Analysis, Evolution, and Reengineering (SANER), 2016 IEEE 23rd International Conference on*, vol. 5. IEEE, 2016. doi: 10.1109/saner.2016.107 pp. 66–77.

[8] S. C. Müller and T. Fritz, "Using (bio) metrics to predict code quality online," in *Proceedings of the 38th International Conference on Software Engineering.* ACM, 2016. doi: 10.1145/2884781.2884803 pp. 452–463.

[9] A. Landowska and M. R. Wróbel, "Affective reactions to playing digital games," in *Human System Interactions (HSI), 2015 8th International Conference on.* IEEE, 2015. doi: 10.1109/hsi.2015.7170678 pp. 264–270.

[10] A. Kołakowska, "A review of emotion recognition methods based on keystroke dynamics and mouse movements," in *Human System Interaction (HSI), 2013 The 6th International Conference on.* IEEE, 2013. doi: 10.1109/hsi.2013.6577879 pp. 548–555.

[11] N. Fragopanagos and J. G. Taylor, "Emotion recognition in human–computer interaction," *Neural Networks*, vol. 18, no. 4, pp. 389–405, 2005. doi: 10.1016/j.neunet.2005.03.006

[12] H. Gunes and M. Piccardi, "Affect recognition from face and body: early fusion vs. late fusion," in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, vol. 4. IEEE, 2005. doi: 10.1109/icsmc.2005.1571679 pp. 3437–3443.

[13] A. Majumder, L. Behera, and V. K. Subramanian, "Emotion recognition from geometric facial features using self-organizing map," *Pattern Recognition*, vol. 47, no. 3, pp. 1282–1293, 2014. doi: 10.1016/j.patcog.2013.10.010

[14] M. A. Sayette, J. F. Cohn, J. M. Wertz, M. A. Perrott, and D. J. Parrott, "A psychometric evaluation of the facial action coding system for assessing spontaneous expression," *Journal of Nonverbal Behavior*, vol. 25, no. 3, pp. 167–185, 2001. doi: 10.1023/A:1010671109788

[15] A. Landowska, "Emotion monitor-concept, construction and lessons learned," in *Computer Science and Information Systems (FedCSIS), 2015 Federated Conference on.* IEEE, 2015. doi: 10.15439/2015f264 pp. 75–80.

[16] M. R. Wrobel, "Applicability of emotion recognition and induction methods to study the behavior of programmers," *Applied Sciences*, vol. 8, no. 3, p. 323, 2018. doi: 10.3390/app8030323

[17] G. Brodny, A. Kołakowska, A. Landowska, M. Szwoch, W. Szwoch, and M. R. Wróbel, "Comparison of selected off-the-shelf solutions for emotion recognition based on facial expressions," in *Human System Interactions (HSI), 2016 9th International Conference on.* IEEE, 2016. doi: 10.1109/hsi.2016.7529664 pp. 397–404.