# Initial evaluation of the brain activity under different software development situations

Rustam Ikramov, Vladimir Ivanov, Sergey Masyagin, Ruslan Shakirov,
Ilyas Sirazidtinov, Giancarlo Succi, Ananga Thapaliya, Alexander Tormasov, Oydinoy Zufarova
*Innopolis University*
Innopolis, Russia
f.last@innopolis.ru

***Abstract*—The use of biological signals to understand software development has become more popular in the last few years but poses new challenges with respect to the overall experimental settings. In this paper we present such challenges and the approach we took to overcome them. We illustrate our approach by evaluating two programming situations: pair programming and programming with music. The subjects involved in the experimentation are mostly students, however, in the largest case we involved graduate students coming from industry with at least three years of working experience. The results in general support the validity of this approach and encourage to go further in this research line. Moreover, as a byproduct, the analysis of pair programming confirms, from a biological perspective, early hypotheses that pair programming induces higher level of concentration.**

***Index Terms*—Empirical methods, software experimentation**

## I. Introduction

Software is the result of the creating activity of software developers. It is clear that improving developers' working conditions might lead to improving the quality of created software the same as productivity of the employer. It is also should be said that working in an area full of distracting agents or activities will decrease the worker's performance. There are lots of different assumptions and myths on how to improve the software development process, but almost no one has an argumented proof. Moreover, there is no a general way for evaluating a developer's physical and mentional state. As a consequence the state of mind of developers play a major role in the quality and the productivity of the produced software systems. In the recent years, the research arena has become more aware of this fact and new studies have emerge, some of which also directly analyzing biological signals. However, the overall research field in this area is still in its infancy. This paper presents the early results and challenges of using a full EEG to understand the brain activity during coding, in research that started about two years ago [1]. Specifically, we are trying to evaluate empirically how different settings may induce different brainwaves, and from this, understand the mental states of developers in such different settings and thus devise the most suited for a variety of work tasks and conditions. In this prototypical phase, a wide approach has been taken in collecting and analyzing the data, considering "standard" working tasks, in essence preferring breadth over depth in the analysis [27].

Our intention is threefold:

- to perform a preliminary observational evaluation of the areas where phenomena could occur for a followup deeper evaluation;
- to gather a better understanding of the opportunities and the problems arising when collecting and analyzing developers data using EEG, in the hope of facilitating future research;
- to expand our research by supplying our initial results to researchers and research groups interested in replicating our findings.

We have considered two settings, primarily because they represented two situations already present in our working context:

- developing using pair programming (the largest part of the experiment);
- developing with music in the background (still considered, given the interest of the involved researchers).

Notice that we have decided to include in this paper also the small portion of collected data referring to programming with music, as it uses a different experimental protocol which adds a significant breadth at this initial investigation.

The subjects involved in our research belong primarily to two groups:

- graduate students with at least three years of working experience in the industry, who can be assimilated to professionals
- undergraduate students

The unique contribution of this work is that the results in general support the validity of this approach and encourage us to go further in this research line. Moreover, as a byproduct, the analysis of pair programming confirms, from a biological perspective, early hypotheses that pair programming induces higher level of concentration; this appears quite remarkable.

This paper is organized as follows. Section II presents the overall background of the paper. Section III outlines the approach taken to analyze the data and how the data was collected. Section IV details the analysis of the data that we have collected. Section V summarizes the early results that we have obtained so far. Section VI outlines the challenges that we have faced in this kind of empirical work to share it with

DOI reference number: 10.18293/SEKE2019-215

other researchers worldwide in the quest of identifying best practices. Section VII draws some conclusions.

TABLE I
SUMMARY OF THE EXPERIMENTS

| Id | Situation | Subjects | N | Analysis |
|---|---|---|---|---|
| 1 | Pair programming | Graduate students with working experience | 10 | ERD |
| 2 | Pair programming | Undergraduate | 3 | Correlation |
| 3 | Programming with music | Undergraduate | 2 | ERD |
| 4 | Programming with music | Undergraduate | 2 | Correlation |

## II. Background

As mentioned, there has been an increased interest in using biological signals to understand the mind of developers, in particular using three main kinds of devices:

- electroencephalogram (EEG),
- functional magnetic resonance imaging (fMRI),
- various bio-metric sensors.

*a) Electroencephalogram:* This is the technique we are considering. To our knowledge, so far a complete portable EEG device has been used in areas related to software engineering only in the study conducted by Lee et al. (2016) [16] on exploring how the mind of developers evolved from novice to experts in program comprehension tasks.

*b) Functional magnetic resonance imaging:* Functional magnetic resonance imaging (fMRI) provides indirect estimation of brain activity, measuring metabolic changes in blood flow and oxygen consumption as a result of increased underlying neural activity. This technique allows the detection of active regions of the brain [8]. As a result, fMRI is widely used to determine specific brain regions which are responsible for the certain mental activity. In order to learn about software developers' brain activity, researchers chose code review and code comprehension as the primary activities for which brain activity needs to be understood [8], [22], [23].

Siegmund et al. (2014) detected activation specific Broadmann-areas during code comprehension [22]. In their followup work (2017) they investigated the difference between bottom-up program comprehension and comprehension with semantic cues in terms of brain areas involved [23]. This study uses very accurate techniques to explore the work of the brain, the fMRI. Floyd et al. (2017) have performed a similar study applying fMRI to understand the mental activities surrounding program comprehension [9].

*c) Ensemble of bio-metric sensors:* An alternative approach has been to use an ensemble of bio-metric sensors like eye trackers for measuring pupil size and eye blinks, electroencephalography to determine brain activity, electrodermal activity sensors to detect skin-related activity, and heart-related sensors [10], [17], [29].

TABLE II
TECHNICAL CHARACTERISTICS OF THE MITSAR SMART-BCI EEG DEVICE

| Options | Smart BCI EEG headset |
|---|---|
| EEG channels | 24 |
| Poly channel | 1 for ECG |
| Reference | A1, A2, (A1+A2)/2, Cz, REF |
| Frequency band | 0(DC) 70 Hz |
| Sampling rate | 2000 Hz |
| Storage rate | 250 Hz |
| Noise | 1.2$\mu$V peak-to-peak |
| Input range | $\pm 300\mu$V |

This approach was applied in a series of investigations which will be described below. The main interest in these investigations was to obtain metrics that correlate with software developers performance. Züger and Fritz (2015) used interruptibility [29] while Müller and Fritz (2015) used positive and negative emotions of software developers [17] as metrics of progress in the change task. They processed the data from multiple bio-sensors and applied methods of supervised learning (Naive Bayes) to distinguish levels of these cognitive states [17], [29].

In these studies, monitoring the state of the mind in depth was limited because:

- the assessment of emotions was performed subjectively by the participants [17];
- a single channel EEG device was used, which may result in an error of up to 50% [21].

## III. Approach to data analysis

*a) Infrastructure used:* We used wireless 24 channel Mitsar SMART-BCI elastic cap for our experiment (details are in Table II). The placement of electrodes was according to the standard 10-20 scheme. Technical characteristics of the Mitsar Smart-BCI EEG device are presented in II. One of the very important steps of EEG recording is the preparation of the EEG cap. We used the canonical type of cleaning before the experiment which is cleaning with spirit. During the data recording, we also used conductive gel to provide a better connection between electrodes and scalp.

Since we use a multi-channel EEG device, the first step to undertake is to select the channels that are the core of the analysis. On the one hand, many channels provide a wide range of information from the whole scalp. On the other hand, this information can be redundant. Moreover, electrodes placed on different parts of the scalp are affected by different types of EGG artifacts, e.g. frontal electrodes are more likely to be affected by muscle and eye movements. During the experimental set up of the device, we found out that a signal from the frontal electrodes cannot be cleaned with EEG prepossessing techniques like Individual Component Analysis and manual filtering. We did not propose any other methods than these two for frontal electrodes since we found out for this particular experiment, central electrodes would be enough for the analysis and result. Based on this fact we decided to analyze only central electrodes (F3, Fz, F4, C3, Cz, C4, P3, Pz, P4) since they provide proper quality data which can be used in further analysis.

The collected data have a lot of interference including:

- imperfection of EEG equipment;
- metal objects nearby;
- Wi-Fi and mobile network, mobile phones;
- artifacts from the person (e.g. blinking, jaw movements, sneeze);
- harsh background sounds;
- size of the cap.

Moreover, there are patterns to take into account, like the age, the gender, and other physiological characteristics of the subject.

Therefore, after the selection of the channels, we have performed a cleaning of the data with the following filters:

- **Amplitude filtering:** All data which was not in the range $[-200\mu V; +200\mu V]$ we considered as an artifact and removed from the signal. If the total share of noisy data in the channel was more than 20% we considered the channel as compromised and removed it from the dataset.
- **High and low pass filters:** The range of filter was picked according to the possible variance of individual alpha and theta waves and equaled to $[2Hz; 15Hz]$.
- A **notch filter** was used to remove the noise from AC lines.

*b) Processing of the EEG data:* As mentioned above, we decided to use only clean channels (data). The choice of clean channels was reasoned by EEG artifacts that are very hard to be recovered to the original data. Moreover, we use use the following infrastructure:

- **Programming tools**: Anaconda 3 Python distribution, NumPy, and SciPy packs, MNE 0.16.1
- **Electrodes**: 'F3-Cz, 'Fz-Cz', 'F4-Cz', 'C3-Cz', 'C4-Cz', 'P3-Cz', 'Pz-Cz', 'P4-Cz' (depending on the setting)
- **Filtering**: Finite impulse response method, as provided by MNE library

Our approach is described in Algorithm 1, implemented, as mentioned using Python 3 with scipy and numpy libraries.

*c) Analysis of the EEG data:* The first step of analysing the data is an adjustment of alpha and theta waves ranges since they could be different for various ages. The variability of alpha waves in age-matched groups has been shown to have a normal distribution ($\mu$ = 10Hz, $\sigma$ = 1Hz) and exhibits tonic changes, increasing from childhood to adulthood, then declining according to the following formula [13]: $PeakAlphaFrequency = 11.95 - 0.053 \times Age$

We computed peak alpha frequency for each participant (or Individual Alpha Frequency - IAF) and used as the anchor point for calculating alpha sub-bands.

The importance of alpha sub-bands comes from the fact that they improve the accuracy of amplitude measures and more accurately reacts on functional differences of the different oscillators, i.e., functional groupings of neurons, which

**Data:** EEG measurements of participants
**Result:** ERD Distributions
**for** *each measurement in Data* **do**
    IAF(individual $\alpha$ frequency) = 11.95 - 0.053 $\cdot$ Age;
    theta = [IAF - 6; IAF - 4];
    L1A = [IAF - 4; IAF - 2];
    L2A = [IAF - 2; IAF];
    UA = [IAF; IAF + 2];
    fft = FFT(measurment) erdall= (calibration (participant) - fft) / calibration(participant);
**end**
**for** *each subband* **do**
    erd [subband] = mean(erdall [subband]);
**end**

**Algorithm 1:** ERD distribution calculating algorithm

contribute to alpha power. For instance, the phasic changes in the lower-1 alpha (L1A) and lower-2 alpha (L2A) subbands are considered to be as an indicator of task-related attentional demands including both components of attention - alertness and arousal [14]. On the other hand upper alpha (UA) changes correlates with semantic memory processing and synchronization in the theta band reflects episodic memory and the encoding of new information [14]. Concluding all above it can be said that our choice of features depended on the connection between the EEG feature and the cognitive processes that this feature can represent.

In our study we used these ranges of sub-bands:

- L1A range is [IAF - 4Hz ; IAF - 2Hz]
- L2A range is [IAF - 2Hz; IAF]
- UA range is [IAF ; IAF + 2Hz]
- Theta range is [IAF - 6Hz ; IAF - 4Hz]

Next step is counting the number of waves included in the corresponding interval. In this way, we can evaluate the brain activity at each time point.

The analysis is then centered in two main techniques:

- ERD,
- Correlations of brainwaves.

The **ERD** (Event-Related Desynchronization) is a measure of the level to which neurons no longer oscillate in synchrony as they become activated to process the given task [5]. Consequently, more task demanding work should cause bigger ERD difference between rest and programming periods. ERD is calculated as it is shown in the formula below:

$$ERD = \frac{(amplitude)_{rest} - (amplitude)_{programming}}{(amplitude)_{rest}} \times 100\%$$

The ERD is computed for 2000ms window of the signal via Fast Fourier Transformation (FFT). As a result, we obtain a time-series or distribution of ERD for each sub-band for each different programming activity. The name convention of the ERD time-series is presented in Table VI.

Intuitively calculating ERD is subtracting the values of the spectrum from calibration value and normalizing on the calibration value. As a result we obtain a normalized spectrum of difference in which we find a mean value for the specific frequency ranges. We performed this procedure for each channel and calculated resulted distributions as the average among all channels. For example, we can have active spectrum only for alpha and theta waves as seen from Table III in case of pair programming. This implies that the result can vary and we can get active spectrum for other different waves based on different ERD value based on different types of experiment.

The analysis of the **correlation of brainwaves** identifies the relationships existing among theta and L1-alpha waves, L2-alpha and upper alpha waves. Strong correlations explain different mental activities and statuses.

For instance from all the data obtained from EEG, individual L1-alpha waves stands out as a measure that can be correlated with other brainwaves such as L2-alpha or upper alpha waves. For example, in our studies correlation between these waves in case of pair programming was slightly higher as compared to solo programming whereas this correlation was lower in the case of programming with music rather than without music. These examples from our study imply that correlation can differ affecting the results to be higher or lower depending on the type of experiments we are performing.

*d) Experimental protocol:* In all cases the students were divided in two groups: treatment and control, even if in one case the control group was very small; again, please remember that the goal of this study is to determine in practice the feasibility of the approach rather than performing sound and reliable observation for the situation under consideration. Each part of the experimentation was scheduled in a separate day and, given the initial availability of two EEG device, when two subjects were involved, they were analyzed together. The following is the detailed steps and here P1 indicates participant one and and P2 indicates participant two.

The steps for the analysis of pair programming have been:

1) Calibrating P1 and P2. The calibration part consists of two parts. First one is when subjects sit with closed eyes in front of the computer in a restful state and the second one is the same but with opened eyes. The steps are required to measure alpha and theta synchronizations during calm state.
2) Solo programming of P1 and P2 (60 minutes).
3) Break, rest period without hard mental activity (10 minutes).
4) Pair programming, P1 is on driver mode, P2 is a navigator (60 minutes).
5) Break, rest period without hard mental activity (10 minutes).
6) Pair programming, P1 is on navigator mode, P1 is a driver (60 minutes).

The steps for the analysis of the effect of music have been:

1) Calibration P1 (with and without music): First, the subject sits with the closed eyes in front of the computer in a calm state and for the second time with the same instructions but with opened eyes. As it was mentioned before, these steps and instructions are necessary to determine the alpha and theta synchronization during the restful state.
2) P1 starts programming for the given task without music. (60 minutes)
3) Rest period without any types of hard mental activity. P1 is on calm state (break) for 10 minutes.
4) P1 starts programming for the given task and listening for a music (the music was chosen by P1 according to his personal preferences). (60 minutes)

*e) Description of the collected data:* As mentioned, the subjects involved in our research belong primarily to two groups:

- volunteer graduate students with at least three years of working experience in the industry, who can be assimilated to professionals,
- volunteer undergraduate students.

The graduate students were mostly recruited during the so-called "bootcamp," a two weeks course preparing our students to of preparation to study. Such students are between 23 and 30 years of age and come directly from industry with at least 3 years of experience, so we can consider them almost as professional for the purpose of the generalizability of data.

The undergraduate students were mostly second year students participating at the data collection for curiosity and interest in neurosciences.

Excluding calibration data, the dataset contains 36 hours of recorded EEG data mostly for the analysis of pair programming (11 hours for driver, 11 for navigator and 12 for solo) and 2 hours for programming with music.

## IV. Analysis of the collected data

Pair programming (PP) is a technique of extreme programming and other agile methods where two developers work together on one workstation, one being the "driver," who uses the keyboard and write the code, the other being the "navigator" who provides systematic guidance to the driver [12]. Pair programming was picked as a primary topic of the study since it may influence on software developer's productivity and attention. There have been multiple studies on pair programming evidencing its pros and cons, the pros including: reducing a defect rate, improving the design, increasing productivity [6], and increased concentration of developers [25]. Music Programming is a common practice but, despite of this, rarely investigated: developers and programmers listen to music of their choice while coding.

Our experiment about Pair Programming involved 11 graduate students with ERD to analyse the data and 3 undergraduate students with correlation analysis; our experiment with Music Programming involved 2 undergraduate students and used correlation to analyse the data (Table I).

*a) ERD:* During the evaluation using ERD, we compare the ERD values in 3 working cases: solo, driver, and navigator. We check the difference between such values using the non-parametric Mann-Whitney test and we determine the significance of the difference. As mention, given the exploratory goals of this paper we do not systematically track the significance of the result; in this case we use the significance level as an indication of a significant effect of the "treatment," that is, working in pair or working with music.

Specifically, we consider ERD of theta waves, which desynchronizes (decreases) with the higher memory load, ERD of all alpha ranges (L1A, L2A, UA) synchronizes (increases) with a higher level of attention and semantic memory processing (n other words, the higher value of ERD in alpha band indicates higher attention and semantic memory processing during the given task for the given participant). Using this information we can calculate statistics of ERD distributions of the same sub-bands but from the different activities and compare them.

TABLE III
VALUES OF ERD IN THE FIRST EXPERIMENT

| Sub-band | Highest value | Significance | Interpretation |
|---|---|---|---|
| L1A | Pair - navigator | Yes | Higher attention required |
| L2A | Pair - navigator | No | As above |
| UA | Not conclusive | No | Nothing |
| Theta | Solo | No | Usually opposite of L1A, so confirms the results |

*b) Correlations:* Using correlations we compare Pearson's correlation coefficients between the 3 cases of pair/solo programming (solo, pair/driver, and pair/navigator) and the 2 cases of programming with and without music. The brainwaves differ from each other while any kind of mental or physical activity is done by the object. As theta waves decrease with the higher memory load and all the alpha ranges (L1A, L2A, UA) increase with a higher level of attention and semantic memory processing, the correlation of this waves should differ over time. Using this information we can calculate the statistics of the correlation of the same sub-bands from the different activities and compare them. To perform a comparison of Correlation, we performed Pearson's correlation coefficients (Tables IV and V).

## V. RESULTS AND DISCUSSION

*a) Analysis with ERD:* In general, desynchronization in the lower alpha band reflects higher levels of attention [14]; for such band in the case pair programming we obtained the highest ERD for pair-navigator mode and equal values for solo and pair-driver mode (Table III). It may mean that pair programming in navigator mode requires more attention, and this reflects the intuition that the navigator position requires evaluating and guiding the development, which in turn intuitively requires a significant effort of attention, also because the navigator is not involved in a physical contact with the keyboard. The analysis of UA was not conclusive.

TABLE IV
CORRELATION ANALYSIS FOR PAIR PROGRAMMING

| Participant | Theta and L1-$\alpha$ | L2-alpha and Upper $\alpha$ |
|---|---|---|
| 1 (PP-Driver) | 0.9 | 0.8 |
| 1 (PP-Navigator) | 0.86 | 0.82 |
| 1 (Solo) | 0.80 | 0.86 |
| 2 (PP-Driver) | 0.799 | 0.9 |
| 2 (PP-Navigator) | 0.81 | 0.85 |
| 2 (Solo) | 0.84 | 0.87 |
| 3 (PP-Driver) | 0.88 | 0.82 |
| 3 (PP-Navigator) | 0.93 | 0.73 |
| 3 (Solo) | 0.875 | 0.81 |

TABLE V
CORRELATION ANALYSIS FOR PROGRAMMING WITH MUSIC

| Environment | Theta and L1-alpha | L2-alpha and Upper alpha |
|---|---|---|
| With music (Participant 1) | 0.825 | 0.878 |
| Without music (Participant 1) | 0.875 | 0.815 |
| With music (Participant 2) | 0.827 | 0.91 |
| Without music (Participant 2) | 0.827 | 0.835 |

According to Klimesch et. al. [14] synchronization in the theta band reflects episodic memory and the encoding of new information. For the theta region we obtained a highest value for solo programming, followed by the navigator, and finally the driver. Theta and alpha waves are supposed to be invariant, which roughly means when alpha increases, theta decreases, and vice versa. As a result, we have that higher desynchronization means lower synchronization. If we denote $ERS$ as event-related synchronization we get the following relation: $ERS_{pair-driver} > ERS_{pair-navigator} > ERS_{solo}$.

Anyway, for now, it is difficult to interpret the meaning of difference in episodic memory working. However, the second part which states the theta band reflects the encoding of new information might be true in case of pair programming.

The analysis of ERD for programming with music did not evidence any specific patterns, perhaps also because of the limited dataset available.

*b) Analysis with correlations:* The analysis of the correlation for pair programming (Table IV) appears somehow to support the claims made with the analysis of ERD. Indeed, the very small dataset is not conclusive for practical reasons, still seeing a second experiment conducted with a different approach hinting at the same pattern as the first one, provides some observational confirmation of the statement that the navigator in pair programming has higher level of attention.

The analysis of the correlation with music (Table V) is again not conclusive, and again we can replicate the limits of the small dataset.

## VI. CHALLENGES ENCOUNTERED

Since the goal of this paper is primarily to provide a reference for future experiences in using biological sensors to detect the states of minds of developers, it is important to underline the different challenges that emerged during the experimentations, so that future research can take suitable precautions to mitigate or even eliminate them:

1) As this was quite a new experiment in the field of computer science, there was a lack of other works and papers related to the field of computer science to structure our overall experimental setting, therefore it took a considerable effort to define a solid experimental protocol and in due course a significant amount of data got lost.
2) The EEG picked up a lot of muscle activity, clouding our data. So subjects had to stay as still as possible and blink as minimum as possible.
3) The device could not record from the subjects with the thick hair even with the addition of the gel.
4) The EEG experiment was highly influenced by environment noise, so a lot of filtering was done. Location of the experiment highly depends on the goal of the experiment, so it was difficult to find its perfect place.
5) Large number of subjects were required and a huge number of experiments were conducted for extraction of useful data and information from the device because the device had poor signal to noise ratio, therefore, this approach is quite effort intensive.
6) It took a long time to start the experiment because the device required a complex arrangement of many electrodes around the head with the use of different gels; moreover, also the setup of the computer software required some time.

## VII. Discussion and Conclusion

As mentioned, the goal of our work is to provide a new contribution to people interested in performing analysis of software development using biological signals, thus discovering a whole new understanding of the state of mind of developers, who are the main resource in the production of software. To this end we have run four experiments, the largest of which involving 10 graduate students with at least three years of programming experience, so with a professional background similar to developers working in companies, thus providing higher credibility to our observational findings. We have run three additional very small experiments with undergraduate students. The subject of the first largest experiment and of a second small experiment was to analyze pair programming, while the other two small experiments focused on programming with music.

The first result that we have obtained is that, despite several possible challenges, some of which discussed in Section VI, the approach appears to work. For the largest experiment, anyway involving only 10 subjects, we did obtain some observational conclusions confirming previous evidence that pair programming increases the level of attention from a clear biological standpoint. We think that this result is remarkable.

For the case of programming with music, we have not been able to achieve any significant result. We are not discouraged by this – it is an effect of the significant amount of work required to run such experiment and we think that a larger experiment may lead to more conclusive statements.

We have also seen that as the time progresses, indeed, we have become more effective in collecting the required data, so there is an important learning phase that, while it cannot eliminate the significant amount of effort required by this approach, still can partially mitigate it. As a lateral comment, we have not identified any pattern in the data we have lost, so we assume that the results that we have obtained in the largest experiment related to pair programming does not suffer of it.

Moreover a growing number of experiments could be relevant in software relevant for safety critical situations, infrastructures, etc. [2]–[4], [7], [24], [28] or during learning phases [11], [18]. It would also be interesting to involve the open source community in sharing personal data [15], [19], [20], [26].

Summing up, based on all the results, our future work will be based on more focused experimentation on specific programming situations using larger datasets of students and then, indeed, trying to move our analysis to the industry. Also we will try to use not only central electrodes but also the frontal electrodes and for the evaluation, not only correlation and ERD but also other available techniques will be used, thus generating more accurate and comparable results. After applying different approaches for EEG data processing it was found that described correlation methods does not provide veridical outcomes for the further analysis so it should not be used for analyzing EEG data. The observed results might be used for identifying the most productive programming techniques. In the future researches we will test other conditions which may have an impact on developer's productivity.

## VIII. Acknowledgments

We thank Innopolis University for generously supporting this research.

## References

[1] S. Busechian, V. Ivanov, A. Rogers, I. Sirazitdinov, G. Succi, A. Tormasov, and J. Yi. Understanding the Impact of Pair Programming on the Minds of Developers. In *Proceedings of the 40th International Conference on Software Engineering Companion*, ICSE-NIER'18, Gothenburg, Sweden, May-June 2018. ACM.

[2] L. Corral, A. B. Georgiev, A. Sillitti, and G. Succi. A method for characterizing energy consumption in Android smartphones. In *Green and Sustainable Software (GREENS 2013), 2nd International Workshop on*, pages 38–45. IEEE, May 2013.

[3] L. Corral, A. Sillitti, and G. Succi. Software development processes for mobile systems: Is agile really taking over the business? In *Engineering of Mobile-Enabled Systems (MOBS), 2013 1st International Workshop on the*, pages 19–24, May 2013.

[4] L. Corral, A. Sillitti, G. Succi, A. Garibbo, and P. Ramella. Evolution of Mobile Software Development from Platform-Specific to Web-Based Multiplatform Paradigm. In *Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Onward! 2011, pages 181–183, New York, NY, USA, 2011. ACM.

[5] I. Crk, T. Kluthe, and A. Stefic. Understanding programming expertise: An empirical study of phasic brain wave changes. 2015.

[6] E. di Bella, I. Fronza, N. Phaphoom, A. Sillitti, G. Succi, and J. Vlasenko. Memory processes, brain oscillations and eeg synchronization. *IEEE Transactions on Software Engineering*, 39:930 – 953, 2013.

[7] E. Di Bella, A. Sillitti, and G. Succi. A multivariate classification of open source developers. *Information Sciences*, 221:72–83, 2013.

[8] B. Floyd, T. Santander, and W. Weimer. Decoding the representation of code in the brain: an fmri study of code review and expertise. In *Proceedings of the 39th International Conference on Software Engineering, ICSE 2017, Buenos Aires, Argentina, May 20-28, 2017*, pages 175–186, 2017.

[9] B. Floyd, T. Santander, and W. Weimer. Decoding the representation of code in the brain: An fmri study of code review and expertise. In *Proceedings of the 39th International Conference on Software Engineering*, ICSE '17, pages 175–186, Piscataway, NJ, USA, 2017. IEEE Press.

[10] T. Fritz and S. C. Müller. Leveraging biometric data to boost software developer productivity. In *Leaders of Tomorrow Symposium: Future of Software Engineering, FOSE@SANER 2016, Osaka, Japan, March 14, 2016*, pages 66–77, 2016.

[11] I. Fronza, A. Sillitti, and G. Succi. An Interpretation of the Results of the Analysis of Pair Programming During Novices Integration in a Team. In *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement*, ESEM '09, pages 225–235. IEEE Computer Society, 2009.

[12] J. Kivi, D. Haydon, J. Hayes, R. Schneider, and G. Succi. Extreme programming: a university team design experience. In *2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era (Cat. No.00TH8492)*, volume 2, pages 816–820 vol.2, May 2000.

[13] W. Klimesch. Memory processes, brain oscillations and eeg synchronization. *International Journal of Psychophysiology*, 24:61–100, 1996.

[14] W. Klimesch. Eeg alpha and theta oscillations reflect cognitive and memory performance: a review and analysis. *Brain research reviews*, 29(2-3):169–195, 1999.

[15] G. L. Kovács, S. Drozdik, P. Zuliani, and G. Succi. Open Source Software for the Public Administration. In *Proceedings of the 6th International Workshop on Computer Science and Information Technologies*, October 2004.

[16] S. Lee, A. Matteson, D. Hooshyar, S. Kim, J. Jung, G. Nam, and H. Lim. Comparing programming language comprehension between novice and expert programmers using EEG analysis. In *16th IEEE International Conference on Bioinformatics and Bioengineering, BIBE 2016, Taichung, Taiwan, October 31 - November 2, 2016*, pages 350–355, 2016.

[17] S. C. Müller and T. Fritz. Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress. In *37th IEEE/ACM International Conference on Software Engineering, ICSE 2015, Florence, Italy, May 16-24, 2015, Volume 1*, pages 688–699, 2015.

[18] W. Pedrycz, B. Russo, and G. Succi. Knowledge transfer in system modeling and its realization through an optimal allocation of information granularity. *Appl. Soft Comput.*, 12(8):1985–1995, Aug. 2012.

[19] E. Petrinja, A. Sillitti, and G. Succi. Comparing OpenBRR, QSOS, and OMM assessment models. In *Open Source Software: New Horizons - Proceedings of the 6th International IFIP WG 2.13 Conference on Open Source Systems, OSS 2010*, pages 224–238, Notre Dame, IN, USA, May 2010. Springer, Heidelberg.

[20] B. Rossi, B. Russo, and G. Succi. Adoption of free/libre open source software in public organizations: factors of impact. *Information Technology & People*, 25(2):156–187, 2012.

[21] I. M. Rytis Maskeliunas, Robertas Damasevicius and M. Vasiljevas. Consumer-grade eeg devices: are they usable for control tasks? *PeerJ*, 4:1–22, March 2016.

[22] J. Siegmund, C. Kästner, S. Apel, C. Parnin, A. Bethmann, T. Leich, G. Saake, and A. Brechmann. Understanding understanding source code with functional magnetic resonance imaging. In *36th International Conference on Software Engineering, ICSE '14, Hyderabad, India - May 31 - June 07, 2014*, pages 378–389, 2014.

[23] J. Siegmund, N. Peitek, C. Parnin, S. Apel, J. Hofmeister, C. Kästner, A. Begel, A. Bethmann, and A. Brechmann. Measuring neural efficiency of program comprehension. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ESEC/FSE 2017, Paderborn, Germany, September 4-8, 2017*, pages 140–150, 2017.

[24] A. Sillitti, A. Janes, G. Succi, and T. Vernazza. Measures for mobile users: an architecture. *Journal of Systems Architecture*, 50(7):393–405, 2004.

[25] A. Sillitti, G. Succi, and J. Vlasenko. Understanding the Impact of Pair Programming on Developers Attention: A Case Study on a Large Industrial Experimentation. In *Proceedings of the 34th International Conference on Software Engineering*, ICSE '12, pages 1094–1101, Piscataway, NJ, USA, June 2012. IEEE Press.

[26] G. Succi, J. Paulson, and A. Eberlein. Preliminary results from an empirical study on the growth of open source and commercial software products. In *EDSER-3 Workshop*, pages 14–15, 2001.

[27] A. Valerio, G. Succi, and M. Fenaroli. Domain analysis and framework-based software development. *SIGAPP Appl. Comput. Rev.*, 5(2):4–15, Sept. 1997.

[28] T. Vernazza, G. Granatella, G. Succi, L. Benedicenti, and M. Mintchev. Defining Metrics for Software Components. In *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics*, volume XI, pages 16–23, July 2000.

[29] M. Züger and T. Fritz. Interruptibility of software developers and its prediction using psycho-physiological sensors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015*, pages 2981–2990, 2015.