The Journal of Systems & Software 178 (2021) 110946

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

Brain and autonomic nervous system activity measurement in software engineering: A systematic literature review ☆

Barbara Weber [a,*], Thomas Fischer [b,1], René Riedl [b,c]

[a] Institute of Computer Science, University of St. Gallen, Rosenbergstrasse 30, St. Gallen, Switzerland
[b] University of Applied Sciences Upper Austria, Wehrgrabengasse 1-3, 4400 Steyr, Austria
[c] Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria

ARTICLE INFO

Article history:
Received 12 May 2020
Received in revised form 15 January 2021
Accepted 7 March 2021
Available online 17 March 2021

Keywords:
Brain and autonomic nervous system activity measurements
Software engineering
Systematic literature review
Electroencephalography (EEG)
Functional magnetic resonance imaging (fMRI)
Heart- and skin-related measurements

ABSTRACT

In the past decade, brain and autonomic nervous system activity measurement received increasing attention in the study of software engineering (SE). This paper presents a systematic literature review (SLR) to survey the existing NeuroSE literature. Based on a rigorous search protocol, we identified 89 papers (hereafter denoted as NeuroSE papers). We analyzed these papers to develop a comprehensive understanding of who had published NeuroSE research and classified the contributions according to their type. The 47 articles presenting completed empirical research were analyzed in detail. The SLR revealed that the number of authors publishing NeuroSE research is still relatively small. The thematic focus so far has been on code comprehension, while code inspection, programming, and bug fixing have been less frequently studied. NeuroSE publications primarily used methods related to brain activity measurement (particularly fMRI and EEG), while methods related to the measurement of autonomic nervous system activity (e.g., pupil dilation, heart rate, skin conductance) received less attention. We also present details of how the empirical research was conducted, including stimuli and independent and dependent variables, and discuss implications for future research. The body of NeuroSE literature is still small. Yet, high quality contributions exist constituting a valuable basis for future studies.

1. Introduction

We use the term NeuroSE to describe a research field in software engineering (SE) that makes use of neurophysiological methods and knowledge to better understand the software development process, as well as its outcome, the software system. Because humans both develop and use software systems, it is clear that a better understanding of the human nervous system – which constitutes the basis of any human perception, thought, emotion, and behavior – is likely to contribute to a better understanding of the SE process and, as a consequence, should also positively affect the software system itself. Neurophysiology is a scientific field that is concerned with the investigation of the functioning of the nervous system, which consists of the brain and other neural tissue in the body (Kandel et al., 2013). The NeuroSE field is relatively young and characterized by collaboration of researchers from various disciplines (e.g., computer science, cognitive neuroscience, and psychology). To the best of our knowledge, the earliest study using brain or autonomic nervous system activity measurements (also referred to as neurophysiological measurements) in an SE context is from 2006 (Aschwanden and Crosby, 2006).

After that inaugural publication, further studies followed, and today a body of literature exists which we characterize as scattered. It follows that a cumulative research tradition does not exist. As we outline in more detail below, for NeuroSE research to progress, a more cumulative tradition is beneficial. A comprehensive review, along with a critical evaluation of the field, constitutes a valuable foundation for the future development of a viable research field. In this article, we present such a review. To the best of our knowledge, such a comprehensive NeuroSE review paper does not exist. The related work that we identified consisted of five related reviews (Goncales et al., 2019; Menzen et al., 2020; Obaidellah et al., 2018; Riedl et al., 2017b, 2020a; Sharafi et al., 2015b), yet none of them had the goal of covering the entire body of NeuroSE literature comprehensively, both from a thematic and from a neuroscience methodology perspective. Specifically, eye tracking research in an SE context is analyzed in the reviews by Sharafi et al. (2015b) and Obaidellah et al. (2018). However, most papers covered by these reviews did not collect neurophysiological data (i.e., eye activity that is largely controlled

☆ Editor: Alexander Serebrenik.

* Corresponding author.
E-mail addresses: barbara.weber@unisg.ch (B. Weber), thomas.fischer@jku.at (T. Fischer), rene.riedl@fh-steyr.at (R. Riedl).

[1] Present address: Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria.

https://doi.org/10.1016/j.jss.2021.110946

by the autonomic nervous system (ANS)), but rather focused on eye movements which are not directly controlled by the ANS, namely fixations and saccades. In another review, Goncales et al. (2019) put an emphasis on one specific construct only, namely cognitive load. Moreover, this review neither analyzed the use of neurophysiological methods comprehensively, nor did it focus on the SE context (rather, this review included several papers that investigated cognitive load in a broader context). Review work recently published by Riedl et al. (2017b, 2020a) investigated studies using neurophysiological methods in the field of Information Systems (referred to as NeuroIS). However, our SLR focuses on software development and not on the use and impact of software systems, the focus of IS research (Sidorova et al., 2006). Most closely related to our work is the review by Menzen et al. (2020), which investigates the usage of biometrics in an SE context (covering both neurophysiological and behavioral measures). However, in their review they did not make a distinction between studies that collected and analyzed biometrics and studies that only describe their potential use. In addition, our SLR covers 89 papers, a much larger database compared to the 40 studies in the paper by Menzen et al. (2020). Despite the fact that our review offers a unique contribution, we stress that it is intended as a complement to the presented related work, and not a substitute for it.

The goal of this SLR is to provide a comprehensive overview of existing research using brain and/or autonomic nervous system activity measurements to investigate SE phenomena. In this literature review we answer the following research questions:

- RQ1: Who published NeuroSE research and where? An answer to this question shows who the most productive NeuroSE authors are and in which outlets they have published. In particular, in an area that is relatively nascent and highly interdisciplinary, such an analysis is valuable since it can help to identify reviewers with NeuroSE experience and editors that could handle NeuroSE submissions. Moreover, it can help researchers who are new to NeuroSE to identify potential collaboration partners, as well as possible publication outlets.
- RQ2: What kind of NeuroSE research has been published?
  - RQ2.1: What is the type of contribution? An answer to this question classifies the studies as "empirical", "research in progress", "methodological", "review", and "conceptual". We argue that for NeuroSE research to progress, a study of contribution type is essential. This may both provide valuable insights into the maturity level of the field and outline critical avenues for the future development of NeuroSE. Specifically, a high rate of completed empirical studies, along with the availability of a relatively large number of review papers and methodological contributions, indicates a relatively high maturity level (Vessey et al., 2002).
  - RQ2.2: Which major thematic orientation did NeuroSE researchers choose? An answer to this question reveals which software development activity (e.g., code comprehension, code inspection, programming) the existing body of literature focuses on.
- RQ3: Which neurophysiological methods and measures were applied in NeuroSE publications? An answer to this question reveals the neurophysiological methods used (e.g., fMRI, EEG, eye tracking), describes the measures that were applied, and explains their usage in answering different types of research questions. Such an analysis can therefore guide future research.
- RQ4: How were the empirical NeuroSE studies designed? An answer to this question reveals various methodological aspects including study population, stimuli, experimental design, dependent and independent constructs as well as data analysis.
- RQ5: What are the main findings of existing NeuroSE research? An answer to this question provides a condensed overview of the main insights that have been obtained thus far based on the application of neurophysiological methods.

The major contributions of this paper are (1) a systematic mapping of the existing NeuroSE literature and (2) a discussion of the implications of the results and an outline of directions for future research. It is hoped that this review instigates a more cumulative research tradition in the future. The remainder of this paper is organized as follows: Section 2 outlines fundamentals of human neurophysiology. The knowledge of human physiology presented in this section is abstracted, as it is intended to serve as a brief introduction to the field for interested software practitioners and mainstream computer scientists and SE researchers. Section 3 describes the research methodology of this review. Results are presented in Section 4. We discuss implications and outline limitations in Section 5. Finally, we present concluding remarks in Section 6.

2. Background

The nervous system constitutes the basis for human perceptions, thoughts, feelings, and behavior; it consists of different parts. At a high level of abstraction, we can distinguish the central nervous system (CNS) and the peripheral nervous system (PNS). The CNS is subdivided into the brain and spinal cord's neural tissue, the PNS comprises all neural tissue except for the CNS. The PNS can be further sub-divided into the somatic nervous system (SNS) and the autonomic nervous system (ANS). The SNS consists of cranial and spinal nerves to and from the sensory organs, muscles, joints, and skin. The main functions of the SNS are the production of movements and the transmission of sensory information (e.g., temperature, touch). The ANS, by contrast, consists of the sympathetic division (which activates the body), the parasympathetic division (which relaxes the body) and the enteric nervous system (which governs the function of the gastrointestinal tract). Based on this overview of the human nervous system, it becomes clear that the brain (i.e., the information processing unit), as well as the sympathetic and parasympathetic divisions of the ANS (which keep the body in balance, referred to as homeostasis), are the major units of analysis in NeuroSE research (Mack et al., 2013; Riedl and Léger, 2016).

Different neurophysiological methods exist to capture neural activity. Functional magnetic resonance imaging (fMRI), electroencephalography (EEG), and functional near-infrared spectroscopy (fNIRS) are important methods to study brain activity. Measurement of heart rate and heart rate variability, electrodermal activity, as well as eye-related measures such as pupil dilation and eye blinks are major methods to study ANS activity. We briefly summarize the main characteristics of these methods in the remainder of this section.

fMRI: A magnetic resonance imaging (MRI) scanner measures blood oxygenation in the brain and exploits the different magnetic properties of oxygenated and deoxygenated blood. Details of this mechanism can be found in the literature on blood-oxygen-level dependent (BOLD) contrast (Kwong et al., 1992). Evidence indicates that the BOLD signal is a good proxy for neuronal activity (Logothetis, 2008; Logothetis et al., 2001), and hence fMRI can be used to investigate the neural correlates of cognitive processes. Note that neurons do not have internal reserves of

energy in the form of oxygen and sugar. It follows that their firing causes a need for more energy to be provided quickly. Based on a process called the hemodynamic response, blood releases oxygen to active neurons at a greater rate than to inactive neurons. The consequence of this process is a change of the relative levels of oxygenated and deoxygenated hemoglobin that can be detected on the basis of their differential magnetic susceptibility. The BOLD contrast, importantly, is sensitive to the presence of deoxygenated hemoglobin (Toga and Mazziotta, 2002). MRI is carried out through a cylindrical tube equipped with an electromagnet that generates field strength (measured in Tesla), which is about 50,000 times stronger than the field strength produced by the earth. Based on fMRI, it is possible to identify activity in a specific brain area within the millimeters range. Thus, spatial resolution is very high. Changes in brain activation that result from stimulus perception can be identified within a few seconds (Riedl and Léger, 2016). Therefore, the temporal resolution of fMRI is in the order of seconds, as is the hemodynamic response lag (for details, please see Table 1). Methodologically, the experimental design of an fMRI study typically involves one of two designs: block or event-related (Riedl and Léger, 2016). In a block design, stimuli pertaining to the same experimental condition are grouped and presented in blocks of time that are separated by resting periods. In an event-related design, stimuli of different experimental conditions are shown in random order.

EEG: Electroencephalograms (EEG) are recordings of the electrical activity of neurons in the brain. Using electrodes placed on the scalp, EEG measures the summation of synchronous postsynaptic action potentials produced by a population of neurons with a very high temporal precision (milliseconds) (Bronzino, 1995). The EEG system is composed of electrodes (which are usually placed on the scalp with a net or cap), amplifiers, an analog-to-digital converter, and a recording device (typically a computer) (Riedl and Léger, 2016). EEG systems amplify and record small voltage fluctuations measured between pairs of electrodes, usually an electrode and a reference electrode. Amplitude of tens of microvolts are typical for EEG studies. The most common analysis of EEG activity is in terms of frequency (Müller-Putz et al., 2015; Riedl and Léger, 2016). Identification of neurocognitive processes specific to a particular event is a challenge, given that EEG measures refer to the summation of the electrical influx of a large number of neurons. To tackle this challenge, the Event Related Potential (ERP) technique was developed. An ERP, or evoked potential, is a patterned fluctuation of voltage recorded by the EEG that represents a cognitive process specific to a discrete event. If the background EEG activity is not filtered, it is hardly possible to identify an ERP signal, because it has low amplitude in comparison to the general EEG signal and other factors such as cardiac activity or muscle contractions that are referred to as "noise" in the EEG literature. Hence, many trials are needed to average responses and to filter the signal (Müller-Putz et al., 2015; Riedl and Léger, 2016). The temporal resolution of EEG is excellent, that is, changes of EEG patterns that result from changes in stimulus perception can be observed within the milliseconds range. For example, the first substantial peaks in the waveform that often occur about 100 ms after stimulus onset are called the P100 and N100 (attributes: positive or negative, 100 ms latency) or the P1 and N1 (indicating the first positive or negative peak) (Müller-Putz et al., 2015, p. 46). Spatial resolution, however, is highly limited. The so-called inverse problem (Helmholtz, 1853), which indicates that an infinite number of source configurations can generate identical surface potentials as measured by EEG, does not allow for an unambiguous identification of the neural generators (i.e., the location of the neural activity within the brain). This explains why localization of brain activity through EEG requires appropriate a priori assumptions about sources and parameters of volume conduction (Michel et al., 2004). It is critical to note that increasingly consumer-grade instruments (e.g., Emotiv Epoc+ 14-channel wireless EEG headset) are used for research purposes. As we outline in Section 5, it is a matter of ongoing discussion whether or not, and if so in which situations, consumer-grade instruments offer reasonable reliability and validity.

fNIRS: Functional Near-Infrared Spectroscopy (fNIRS) is a brain imaging technique that (like fMRI) uses hemodynamic responses to indirectly capture neuronal activity. However, compared to fMRI, fNIRS is less expensive and more portable, offering higher ecological validity (Riedl and Léger, 2016). Moreover, unlike fMRI, fNIRS is an optical imaging technique that uses near-infrared spectroscopy to detect cerebral blood flow and hemoglobin oxygenation level changes (Bunce et al., 2006; Villringer and Chance, 1997). The proper functioning of neurons relies on the oxygen and glucose supply provided by cerebral blood flow. Brain activity reduces local oxygen and glucose concentrations. Hence, the neurovascular coupling mechanism will increase blood flow in this region, supplying it with the appropriate concentration of the constituents needed to metabolize energy (Riedl and Léger, 2016). This phenomenon is the type of event that fNIRS captures at different points in time in order to assess changes that are a function of different experimental conditions (Bunce et al., 2006; Villringer and Chance, 1997). The common apparatus is composed of light sources applied to the scalp and light detectors sensitive to the light that is reflected by the different components of the cerebral cortex (Riedl and Léger, 2016). Given that a higher concentration of oxygen is needed in brain areas activated by an experimental task, the functional map provided by fNIRS informs the researcher on the different functionalities of brain areas (Bunce et al., 2006; Gefen et al., 2014; Villringer and Chance, 1997). Due to the physics of light propagation and corresponding propagation loss, application of fNIRS to study brain mechanisms has spatial resolution limitations. It follows, then, that light intensity is attenuated in tissue and therefore penetration depth is limited to the first 2–3 cm of the cortex (Bunce et al., 2006; Villringer and Chance, 1997). Thus, compared to fMRI, fNIRS offers inferior spatial resolution and limited penetration depth. However, when it comes to temporal resolution fNIRS is substantially faster than fMRI (sampling rates are typically 1 Hz-10 Hz, note that Hz is equivalent to cycles per second). Yet, the hemodynamic response lag is a few seconds.

Table 1 provides a comparison of the above-described imaging methods along different dimensions.[2]

Heart-related measurements: Heart rate (HR) is typically measured by an electrocardiogram (ECG). Application of ECG includes placing a cathode electrode beneath the right clavicle, a ground electrode under the left clavicle, and an anode electrode on the left side of the abdomen (Riedl and Léger, 2016). Note that HR can also be measured with photoplethysmography (PPG) sensors (placed at the wrist or finger, Rajala et al. (2018)), which use a light-based technology to sense the rate of blood flow as controlled by the heart's pumping action (Elgendi, 2012). Thus, PPG is an optical approach for measuring the blood volume pulse (BVP). However, ECG sensors detect electrical activity produced by a heartbeat. A heartbeat consists of a P wave that refers to

[2] Table 1 is based on https://medizinio.de/en/medical-equipment/mri[(a)], https://imotions.com/blog/eeg-headset-prices/[(b)], https://plux.info/kits/438-fnirs-pioneer-820201240.html[(c)], Bunce et al. (2006)[(d)], Lystad and Pollard (2009)[(e)], Quaresima and Ferrari (2016)[(f)] and Müller-Putz et al. (2015)[(g)]. Note[(h)]: Spatial resolution benefits from ultra-high field strength, today usually 7T scanners. Most studies in the extant cognitive neuroscience literature published in the past decade use 3T scanners. Thus, spatial resolution can even be lower than 2 mm if 7T scanners are used (for a recent example, see Rutland et al. (2019)). All fMRI studies (completed research) in this current NeuroSE review used a 3T scanner.

Table 1
Comparison of three major brain research methods.

	fMRI	EEG	fNIRS
Measure of neuronal activity	Indirect Neural activity is inferred based on magnetic properties of oxygenated and deoxygenated hemoglobin in the brain	Direct Measurement of electrical activity on the scalp, which constitutes the manifestation of the activity of populations of neurons in the brain	Indirect Neural activity is inferred based on oxygenated and deoxygenated hemoglobin concentration changes in the cortical tissue of the brain
Confined space (study participant)	Yes	No	No
Cost of equipment	900,000–4 mil US$[a]	100 US$ (1 channel) to 25,000 to 90,000+US$ (up to 256 channels) for research grade[b]	Starting at 1,200 US$ (1 channel)[c] to 10,000 US$ to 300,000+US$ for research grade[d]
Cost per participant (estimated)	300–500 US$	50 US$ (mainly depends on number of electrodes and whether a research-grade or consumer-grade system is used)	50 US$
Spatial resolution	Excellent (2 mm)[e,h]	Good (10 mm)[e] (based on source localization techniques, strongly depends on the number of electrodes)	Reasonable (∼1 cm)[d] (but only activity that occurs in the first 2–3 cm of the cortex)
Temporal resolution	a few seconds[e]	milliseconds[e]	milliseconds to 1 s[f] (sampling frequency typically 1–10 Hz; up to 100 Hz)
Response lag	Order of seconds[e] (hemodynamic response lag)	Order of milliseconds[g]	Order of seconds[f] (hemodynamic response lag)

atrial depolarization of a QRS complex, which represents ventricular depolarization and contraction of the large ventricular muscles, and of a T wave that reflects the rapid repolarization of the ventricles (Riedl and Léger, 2016). Compared to PPG, ECG is a more direct measurement of heart activity as it directly captures electrophysiological signals that result from the heartbeats, rather than downstream effects such as blood properties at distant locations such as wrist or finger. It follows, then, that while ECG measurement is accurate at the level of milliseconds, PPG based measurement is not due to the delay that is influenced, among others, by the pulse wave velocity and the vascular path from the heart to the location of the PPG sensor among other factors (Lekkala and Kuntamalla, 2017); consistent with this fact, high-frequency components of the signal are attenuated due to the long distance the blood has to travel through the body before being measured at the distant body locations (e.g., Buxi et al. (2015)). Most recent research shows that today it is even possible to use non-contact imaging of peripheral hemodynamics (such as blood volume pulse) to study cognitive and emotional constructs such as stress (McDuff et al. (2020)).

Respiration rate (i.e., breathing frequency) is closely related to heart rate, and can also be very informative for SE researchers, primarily because increased mental activity implies greater consumption of oxygen which, in turn, affects respiration and heart rate. Increases in mental activity typically imply increases in respiration and heart rate (Riedl and Léger, 2016). Heart rate variability (HRV) is predominantly a function of ANS activity. The sympathetic part, among others, increases the heart's contraction rate and force (cardiac output), but decreases HRV. Conversely, the parasympathetic part reduces the heart rate, but increases HRV. This interplay between the sympathetic and parasympathetic parts of the ANS constitutes the physiological basis for the heart's instantaneous response to different situations and needs (Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology, 1996). It follows that a low HRV is undesirable, while a high HRV is desirable. Major HRV indicators, also referred to as features of the signal, are: SDNN: The standard deviation of NN intervals in the signal (note that NN interval refers to the time between two consecutive heartbeats, measured in milliseconds). SDANN: The standard deviation of the averages (taken over specific time segments) of the NN intervals in the signal. RMSSD: This feature depends on the differences of subsequent NN intervals. Square these differences, then take the square root of the arithmetic mean of these squares. LF-HRV: The power of the signal in the low-frequency spectrum (0.04–0.15 Hz). HF-HRV: The power of the signal in the high-frequency spectrum (0.15–0.5 Hz) (Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology, 1996). The first three HRV features are from the time domain, measured in milliseconds, while the remaining two are from the frequency domain, measured in squared milliseconds.

Skin-related measurements: Electrodermal activity (EDA) is a property of the human body that causes continuous variation in the skin's electrical characteristics. A galvanometer is used to assess the degree to which the skin permits transmission of an applied current, and the conductance is influenced by the galvanic state of the skin at different moments in time (Naqvi and Bechara, 2006). EDA reflects two types of activities: tonic and phasic (Boucsein et al., 2012; Naqvi and Bechara, 2006). Tonic activity is typically expressed in units of skin conductance level (SCL) and refers to a smooth and slowly changing level on a time scale of tens of seconds to minutes. Phasic activity is usually expressed as electrodermal response (EDR) or skin conductance response (SCR) and refers to short-lasting changes in EDA that appear as a response to a particular stimulus. Evidence indicates that the frequency of non-specific (i.e., non-stimulus related) changes in SCL is typically 1–3 per/min during rest and over 20 per/min in

high arousal situations (Boucsein et al., 2012; Braithwaite et al., 2015; Dawson et al., 2016). For the decomposition of the EDA signal into a tonic and a phasic component several algorithms have been proposed (for an overview, see Posada-Quintero and Chon (2020)). EDA is related to cognitive, emotional, and attentional states, and it is simple to use, completely non-invasive, and provides data that is easily attributable to a single stimulus if EDR or SCR is used (Riedl and Léger, 2016). Physiologically, a galvanic skin response is a result of changes in the sympathetic part of the ANS. Such changes often occur when an individual is facing specific events and situations, including novelty, anticipation of an outcome, decision making, loud noises, fear, or surprise (Riedl and Léger, 2016), and they occur between 1–3 s after stimulus onset (Dawson et al., 2016). Generally, because changes in arousal often do not reach people's awareness level, the use of EDA to capture arousal responses is a popular method (Riedl and Léger, 2016).

Eye-related measures: Oculometry concerns the biometric measurement of the condition and movements of the eye (Riedl and Léger, 2016). Many micro-movements of the eye and pupil-size modifications occur without conscious awareness, also because they are a function of ANS activity rather than deliberate thoughts in the brain. A tool used to measure conditions and movements of the eyes is eye tracking. Researchers applying this tool know, at any moment in time, the target of the gaze, which according to the eye-mind hypothesis is correlated with visual attention and hence it is assumed that we also know what information is being processed by the participant (e.g., on a computer screen) (Just and Carpenter, 1980). Knowing what the eyes are focusing on provides information about what the brain is processing (Riedl and Léger, 2016). To measure the point-of-regard, most systems apply the pupil-corneal reflection method, which uses an infrared camera to locate the features of the eye (Holmqvist and Andersson, 2017). One of the dominant aspects reported in eye tracking analysis is fixation. When the gaze temporarily stops on a specific stimulus (from 200 ms up to several seconds), it is possible to capture the time spent in a specific position (e.g., on a computer screen). Another measure is the saccade. Saccades are the fastest movements made by the human eye, moving from 30 to 80 ms, so that conscious and deliberate information encoding is hardly possible during the movements (Holmqvist and Andersson, 2017). Yet, saccades provide information on search behavior, along with rudimentary color detection and lightweight shape detection. Duchowski (2017) comprehensively summarizes the physiological, methodological and technological characteristics of eye tracking. Moreover, in stressful situations, the sympathetic division of the ANS becomes active and stimulates a number of responses, including pupil dilation (i.e., increased visual attention). When parasympathetic activation occurs, pupils constrict (Riedl and Léger, 2016). In this context, it is critical to mention that pupil dilation, along with more complex related indexes (Duchowski et al., 2020), can be used to measure cognitive load. For a recent review, see Wel and Steenbergen (2018). Finally, eye blinks are also important measures in eye tracking research (Kanoga et al., 2016; Walla et al., 2015). Three blinking types exist: intentional (e.g., if one decides to close the eyes to avoid external stimulation), spontaneous (e.g. corneal lubrication), and reflexive (e.g., Nakano et al. (2012) and Sforza et al. (2008)). Specifically, the startle eye-blink is a reflexive response that typically occurs when an individual encounters a sudden and unexpected stimulus (e.g., loud noise or increase in light). The startle reflex is influenced by brain and ANS activity and can be used to infer affective processing in humans (Lang et al., 1990; Walla et al., 2015).

It has to be noted that we focused on fMRI, EEG, fNIRS, heart rate measurement, electrodermal activity, and eye-related measures because, as it turned out after our review of the SE literature, that these methods were applied frequently in the reviewed studies. In neuroscience and related disciplines, however, a number of further methods are described (Senior et al., 2009). For a description of how each of these tools functions and of common data processing steps required to enable someone who is not familiar with that specific technology to make use of it, we refer the reader to Newman (2019), Harmon-Jones and Beer (2009), and Senior et al. (2009).

3. Methodology of the literature review

In order to identify NeuroSE publications, we conducted a literature search and considered peer-reviewed journal and conference publications. The review process was based on existing recommendations for conducting literature reviews (Kitchenham and Charters, 2007; vom Brocke et al., 2009; Webster and Watson, 2002). Therefore, we first identified keywords based on landmark publications and then selected our outlets for the literature search phase. Based on initially selected papers, we conducted an initial review, followed by backward snowballing, another preliminary review and forward snowballing and finally merged papers with a high degree of overlap referring to the same empirical study (cf. Fig. 1).

3.1. Search strategy

Keywords for the literature search were mainly derived from landmark publications that offer an introduction to the field of NeuroIS (Dimoka et al., 2012; Riedl et al., 2010a; Riedl and Léger, 2016). NeuroIS (Neuro-Information-Systems) "relies on neuroscience and neurophysiological knowledge and tools to better understand the development, use, and impact of information and communication technologies. NeuroIS seeks to contribute to the development of new theories that enable possible accurate predictions of IS-related behaviors, and the design of information systems that positively affect economic and non-economic variables (e.g., productivity, satisfaction, adoption, well-being", (Riedl et al., 2010a); for further details see www.NeuroIS.org). Considering the NeuroSE definition in the Introduction, we argue that NeuroIS is the thematically closest research field with the prefix "neuro". Hence, we based our initial search on landmark publications from this field.

Similar to Riedl et al. (2017a, 2020a) we used terms that are representative of the data collection methods that are highlighted in these landmark publications such as "eye tracking" or "cardiovascular" being representative of cardiovascular measurements. In addition, we combined each "neuro" term with SE terms (e.g., source code or software design) derived from SE handbooks (Sommerville, 2010; Topi and Tucker, 2014). A list of the used "neuro" terms and SE terms used is provided in Table 2. We searched for keywords in title and abstract.

4 Literature databases (journal publications): We searched ACM, IEEE Xplore, Science Direct, Web of Science from 02/05/2018 to 02/08/2018, on 10/26/2018 and from 07/06/2020 to 07/07/2020 using a combination of one neuro term and one SE term at each time without restricting the publication date. This method resulted in 4312 hits.

12 Conferences: Moreover, we searched the following conferences: International Conference on Software Engineering (ICSE), International Conference on Program Comprehension (ICPC), Requirements Engineering: Foundation for Software Quality (REFSQ), Empirical Software Engineering and Measurement (ESEM), European Software Engineering Conference (ESEC), Conference on Human Factors in Computing Systems (ACM CHI), Eye Movements in Programming (EMIP), Emotion Awareness in Software Engineering (SEmotion), Cooperative and Human Aspects of Software Engineering (CHASE), SIGCSE Technical Symposium on

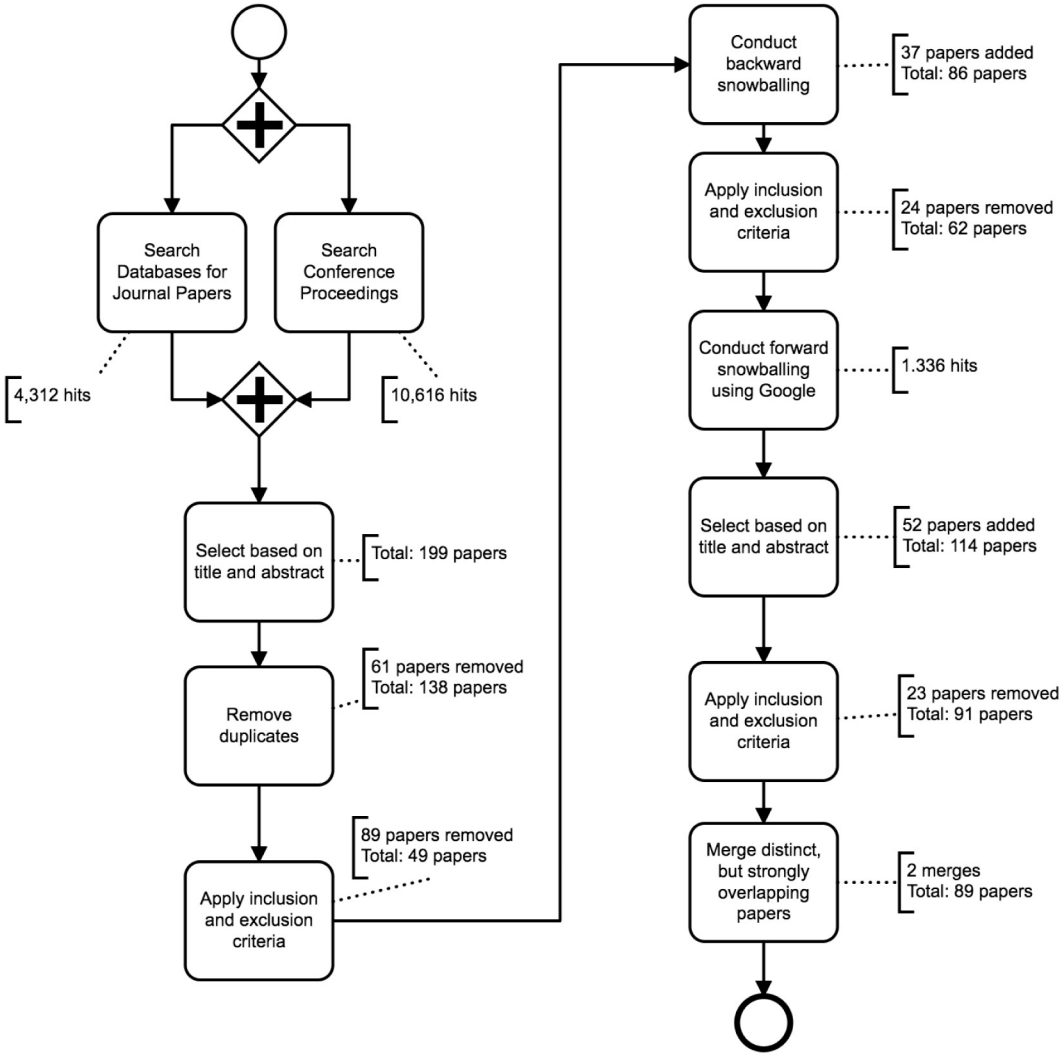


Fig. 1. Overview of the Search Process.

Table 2
Keywords used for literature search.

Neuro terms		
Blood	Electromyography	Morphometry
Brain	Eye tracking	Nervous system
Cardiovascular	fMRI	Positron emission
Cognitive load	fNIRS	Saliva
Cognitive processes	Gaze	Skin conductance
Diffusion tensor	Heart rate	Urine
Electrocardiogram	Hormone	
Electroencephalography	Infrared spectroscopy	
SE terms		
Code review	Program comprehension	Software engineering
Debug	Programming experience	Software testing
Refactoring	Software design	Source code
Requirements elicitation	Software development	
Requirements engineering	Software programming	

Computer Science Education (SIGCSE), International Computing Education Research Conference (ICER), and the International Symposium on the Foundations of Software Engineering (FSE). We used the keywords described in Table 2 or, if no search was available, manual search based on title and abstract from 02/05/2018 to 02/08/2018, on 10/26/2018 and from 07/06/2020 to 07/07/2020 without restricting the publication date. This resulted in 10,616 hits.

After the filtering strategy was applied and an article was selected for inclusion (cf. Section 3.2), we used backward snowballing followed by another review and forward snowballing (cf. Section 3.3).

3.2. Filtering strategy

We first removed totally unrelated papers based on title and abstract which left us with 199 papers (i.e., 157 articles from the database search and 42 articles from the search in the conference proceedings). We then also removed duplicates, which left us with 138 unique papers. The remaining 138 papers were then analyzed in-depth based on the full-text and we applied the following inclusion and exclusion criteria:

- *Inclusion Criterion IC:* The article focuses on the application of neurophysiological methods and/or knowledge to investigate the software development process, and/or its outcome, the software system.
- *Exclusion Criterion EC1:* Main focus on the development of new measurement techniques or the improvement of existing measurement techniques based on neurophysiological data with little or no emphasis on the SE process. For example, papers which proposed new brain–computer interface technology (e.g., Huang and Tognoli (2014)) or self-adaptive systems (e.g., Huang and Miranda (2015)) were excluded.
- *Exclusion Criterion EC2:* Papers focusing on the creation of SE artifacts which integrate neurophysiological data (i.e., neuro-adaptive system), and not on SE-related phenomena (i.e., the investigation of the development process of the SE artifact itself), were excluded (e.g., Riseberg et al. (1998), Scheirer et al. (2002)).
- *Exclusion Criterion EC3:* Papers focusing on the measurement of user experience, if it is not directly related to the SE artifact, were excluded (e.g., Jimenez-Molina et al. (2018)) or where the neurophysiological data stems from the end user of the software and not the developer (e.g., Lin and Imamiya (2006), Phukan (2009)).
- *Exclusion Criterion EC4:* Similar to Riedl et al. (2020a), articles applying eye tracking measurements that are *not* predominantly reflexive (e.g., gaze and saccade measurement) were excluded (e.g., Goldberg (2012, 2014)). It follows, then, that we only included eye tracking studies in this review if their focus was the investigation of pupil dilation or eye blinksstartle reflex.

After applying these inclusion and exclusion criteria to the initially identified 138 papers, 49 remained for further analysis. 14 out of 138 papers did not meet the inclusion criterion and were thus not considered. 6 papers were excluded based on exclusion criterion EC1, 0 based on EC2, 6 based on EC3, and 69 based on criterion EC4.

3.3. Backward snowballing and forward snowballing

This selection was then used for backward snowballing (i.e., searching the references), which resulted in an additional 37 publications being identified. After the application of inclusion and exclusion criteria 24 papers were removed (10 due to IC, 3 based on EC1, 3 based on EC2, 6 based on EC3, and 7 based on EC4), resulting into 13 remaining papers. Based on the previously identified 62 publications, forward snowballing (i.e., tracking the citations) using Google Scholar from 02/05/2018 to 02/08/2018, on 10/26/2018, and from 07/06/2020 to 07/07/2020 was conducted. This resulted in 1336 hits out of which 52 publications were selected for further investigation based on title and abstract (resulting in a total of 114 papers). After the application of inclusion and exclusion criteria 23 papers were removed (7 due to IC, 1 based on EC1, 1 based on EC2, 19 based on EC3, and 29 based on EC4). Therefore, 91 publications constitute the final sample for our literature review. There is a list of selected NeuroSE publications in Appendix A.

3.4. Merging of papers

Finally, we merged overlapping papers referring to the same empirical study. In particular, we merged Fakhoury et al. (2018, 2020) as well as Doukakis (2019) and Doukakis et al. (2020), considering the more comprehensive paper version for further analysis. As a result of this merging, we obtained 89 publications that were included in our SLR.

3.5. Data extraction strategy

To answer the posed research questions (RQs) we extracted the following data.

0. General Info: General information about the paper, i.e., title, authors, outlet, type of outlet (e.g., conference, journal), and publication year. This data was used to answer RQ1.

1. Contribution: To classify the contribution of each publication, in line with Riedl et al. (2020a) we used one of five categories. *"Empirical"* papers focus on testing the relationship between at least two variables and feature information on their study design including data collection and analysis procedures as well as the results of their investigation. *"Research in progress"* papers are also empirical in nature, but do not offer all components of an "empirical" paper, for example, only reporting on preliminary results, or presenting their study design without having completely analyzed or even collected data. *"Methodological"* papers present information on new or existing methodological approaches for NeuroSE research, such as the introduction of eye tracking measures to investigate visual effort (including pupil dilation and blink rates) in the context of different software engineering tasks (e.g., Sharafi et al. (2016)) or the assessment of different emotion recognition methods in terms of their suitability for monitoring the emotional states of developers (e.g., Wróbel and Wrobel (2018)), in some cases supplemented by an exemplary empirical study (e.g., Peitek et al. (2018d)). *"Conceptual"* papers present a discussion on potential constructs for NeuroSE research, related research models (e.g., Brown et al. (2018)), or the design of a new SE artifact. Finally, *"Review"* articles focus on the analysis of previous research, based on a review of the literature (e.g., Sharafi et al. (2015b)). RQ2.1 was answered based on this data.

Moreover, for each completed empirical study we further extracted the following data:

2. Software Development Activities: We classified papers based on the following software development activities: code comprehension, code inspection, programming, change task, bug fixing, documenting code, general. We classified a paper as "code comprehension" when it was about reading and understanding source code snippets (e.g., Siegmund et al. (2014)). The label "code inspection" was used for papers that required detection of errors in the source code going beyond pure syntactical errors (e.g., Castelhano et al. (2018)) or make a decision on whether or not changes should be approved (e.g., Floyd et al. (2017)). We classified a paper as "programming" when it was about writing source code (e.g., Yamamoto et al. (2016)). Papers that required participants to change existing source code (e.g., to add additional functionality) were labeled as "change tasks" (e.g., Müller and Fritz (2015)). In turn, "bug fixing" was used to refer to papers that required the detection and correction of bugs (e.g., González et al. (2015)). Papers that were concerned with documenting existing source code were labeled as "documenting code" (e.g., González et al. (2015)). Finally, the category "general" refers to papers that look at software development as a whole rather than a specific activity (e.g., Müller and Fritz (2016)). Please note that several of the fMRI studies used a code inspection task (more specifically a syntax task) as contrast, while the main focus of interest was

code comprehension. In these cases, we did not label the paper as "code inspection" (e.g., Siegmund et al. (2014)). RQ2.2 was answered based on this data.

3. Methods and Measures

3a. Methods: In this section, we indicate which types of neurophysiological data are collected in completed empirical studies. We consider fMRI, EEG, and fNIRS (the three methods for brain activity measurements), as well as HR, Skin, Eye Tracking, and Other. "Other" is a category that is used to indicate types that have been used infrequently (less than 3 times), such as breath-related measures.

Please note that we excluded studies tracking eye movements that are not directly controlled by the ANS in accordance with Exclusion Criterion 4. It follows that we considered neither those studies that only collected and analyzed fixation-based measures nor those studies based on saccadic measures. Since the focus of this literature review is on neurophysiological data, we only marked a study as "eye tracking" when the study mentioned pupil data or eye blinks (as they are strongly related to ANS activity). Additionally, note that several studies collected EMG (electromyographic) or EOG (electrooculographic) data for artifact removal from the EEG signal. Since this data was not the thematic focus of this paper, we did not consider it here.

3b. Collected measures: We highlight for each method which types of measures are used (e.g., types of skin-related measures, such as skin conductance level or skin conductance response) and how they are measured (e.g., if HR-related data is collected whether the data has been collected using ECG or PPG).

3c. Measurement instruments: Indicates which measurement instruments were used for each of the methods applied (e.g., the device that was used for data collection such as a Tobi TX300 eye tracker or an Emotiv EPOC EEG device).

RQ3 was answered based on this data.

4. Study Participants: provides details concerning the study population

4a. Sample size: Indicates how many individuals participated in the study and also (if applicable) highlights in which part of the study they participated. For example, in some cases several studies are reported in one publication with their respective sample sizes and in some cases different samples have been used for each type of data collected.

4b. Gender: Indicates the gender distribution in the sample (male, female).

4c. Age: Indicates the age distribution in the sample (i.e., chronological age).

4d. Study population: Indicates the occupation of participants (e.g., students or individuals of a certain profession).

4e. Background: Specifies (if indicated) the level of experience of the participants and lists further details (if available) that give some indication of the participant's background.

5. Stimuli: provides details concerning the stimuli used

5a. Task characteristics: provides details concerning the tasks.

5b. Programming language: specifies the programming language in which the stimuli were represented.

5c. Size: indicates the size (in lines) of the code snippets used.

6. Experimental Design: provides details concerning the experimental design

6a. Setting ("Laboratory/Field"): We classified whether an empirical study collected data in a controlled environment ("laboratory") or in a context that is natural to the study population ("field").

6b. Manipulation design ("between-subject/within-subject"): We indicate whether participants experienced all, or at least several, of the conditions of an independent variable ("within-subject") or only one ("between-subject"), or whether it was "mixed". We indicate studies without different experimental conditions with "no conditions".

6c. Experimental procedure: We provide details concerning the experimental procedure. In particular, we mention the presence of repeated measurement (either under different conditions or the presence of multiple trials for the same condition at different points in time).

7. Constructs and Research Questions

7a. Constructs: Includes a list of variables that were investigated in the respective studies. Variables were included here if they were actually measured, as highlighted mostly in the "Methods" section of a publication.

Further, we offer a classification for the type of variable, indicating the relationship between involved variables. The categories are independent variable ("IV") and dependent variable ("DV"). If available, we based this classification on a study's research model or hypotheses. Per default, if a research model or hypotheses were not available, we assumed that manipulated variables were IVs, while measured responses were DVs. When the publication followed a data-driven approach using machine-learning, we assumed the input variables (i.e., the features) to be IVs and the predicted variables to be DVs.

7b. Research questions and hypotheses: lists research questions (and if available hypotheses) investigated in the studies. If not mentioned explicitly in the paper, research questions and hypotheses were reconstructed from the analyses actually conducted in the paper.

7c. Relationship between constructs: states which constructs were used to answer a certain research question or to test a specific hypothesis. Causal relationships are indicated as "IV=>DV", while correlations are indicated as "DV<=>DV".

8. Data Analysis: Indicates the statistical methods used to test hypotheses or to investigate the stated research questions. In some cases, statistical tools used for data cleaning are also mentioned, although we did not focus on data preparation (e.g., artifact removal) at this point.

RQ4 was answered based on this data.

9. Main findings: This section includes summaries of the research findings, predominantly those related to neurophysiological data. Findings as reported in our review are those related to the research questions and hypotheses investigated in the paper.

RQ5 was answered based on this data.

4. Results

This section presents major findings of our literature review structured along our four research questions.

4.1. Who published NeuroSE research and where (RQ1)?

Based on the analysis of N = 89 NeuroSE publications, we identified 191 different authors. The average number of authors per publication is 4.25, and the maximum number of authors is 12. Specifically, we found the following results: 8 papers had 1 author (abbreviated: 8P/1 A), 17P/2 A, 15P/3 A, 13P/4 A, 12P/5 A, 9P/6 A, 6P/7 A, 2P/8 A, 5P/9 A, 1P/10, and 1P/12. Another finding of our analysis is that out of the 191 different authors, 26 researchers (∼14%) authored at least 4 publications and together 42% of all publications. Furthermore, 14 researchers (∼7%) published at least 5 papers and were involved in 28% of all publications (cf. Fig. 2). Fig. 2 shows the concentration of NeuroSE publications across authors. Based on our dataset, we calculated the Gini coefficient (GC), a popular measure of inequality. GC is 0.39; GC=0 expresses perfect equality, where all authors would

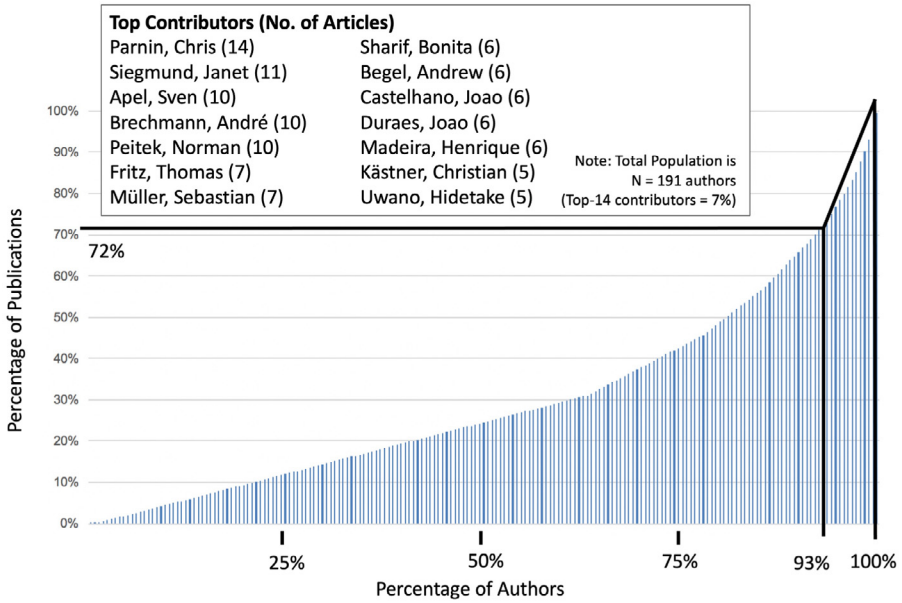


Fig. 2. Concentration of NeuroSE Publications across Authors and Top-7% Contributors.

have contributed an equal number of publications to the NeuroSE literature, and GC=1 expresses maximal inequality.

Based on the analysis of N = 89 NeuroSE publications, we identified 13 outlets where more than 2 NeuroSE papers were published. Most papers appeared at ICSE (11)[2], EMIP (10), ICPC (7), SEmotion (6), and FSE (5) as shown in Table 3.

4.2. What kind of NeuroSE research was published (RQ2)?

This section classifies all 89 NeuroSE papers in terms of their contribution and for the completed empirical research papers we also elaborate on their thematic orientation.

4.2.1. What is the type of contribution (RQ2.1)?

Our analysis revealed the following results for the 89 NeuroSE publications: 47 publications are completed empirical studies (52.81%), 24 publications are research in progress (26.97%), 8 publications are methodological (8.99%), 5 publications are conceptual (5.62%), and 5 publications are reviews (5.62%) (for details, see Appendix C). Moreover,

Fig. 3 illustrates the development of NeuroSE research over time (from 2006 until 2020). The first NeuroSE paper was published in 2006. Until the end of 2013 only three additional papers were published. Starting in 2014 the number of NeuroSE publications began to increase substantially with a peak of 19 publications in 2018.

4.2.2. Which major thematic orientation did NeuroSE researchers choose (RQ2.2)?

This section summarizes the thematic orientation of the completed empirical research papers (N=47). An overview is provided in Table 4. Our results show that existing NeuroSE research has a strong focus on code comprehension (30 out of 47). For example, Siegmund et al. (2014) collected fMRI data while participants were processing short source code snippets to map the brain regions that are active during code comprehension. Code inspection tasks were in the focus of 6 papers (e.g., Floyd et al. (2017)) looked into differences in brain activation between code comprehension and code inspection, while programming tasks were addressed in 6 studies (e.g., Yamamoto et al. (2016) used EEG to predict whether or not a programmer found an implementation strategy during a programming task). Moreover, change tasks were picked up by 3 papers. For example, Müller and Fritz (2015) predicted developers' affective states and the perceived progress during change tasks using multi-modal measurements (i.e., EEG, eye tracking, skin- and heart-related measurements). In addition, bug fixing as well as documentation were picked up by 1 paper only in each case. For example, González et al. (2015) used a consumer-grade EEG tool to compare the signals of participants when documenting code versus bug fixing and programming. Moreover, 5 papers were classified as general. For example, Züger and Fritz (2018) collected heart-related measurements in a field study with professional developers to predict interruptibility during general software development activities.

4.3. Which methods and measures were applied in NeuroSE publications (RQ3)?

This section summarizes which methods and measures were applied in NeuroSE publications and provides examples of typical research questions addressed by the different methods.

4.3.1. Overview of neurophysiological method usage

Fig. 4 shows the extent to which different neurophysiological methods have been used in completed empirical NeuroSE research (N=47).

Methods related to brain activity measurement have been used frequently. Specifically, EEG was used 20 times, fMRI 10 times, and fNIRS 4 times. Regarding ANS activity measurement, we found the following: eye tracking was used 10 times, heart-related measurements 10 times, skin-related measurements 8 times, and we counted one use of a measurement in the category "Other". Moreover, we found that 38 studies used one neurophysiological method only, while 3 studies combined two neurophysiological methods, 5 studies combined three neurophysiological methods, and 1 study combined four neurophysiological methods. For example, Fritz et al. (2014) combined EEG with eye tracking and skin-related measurements. Moreover, Müller and Fritz (2016) additionally used heart-related measurements.

14 studies complemented neurophysiological methods with measurements of behavioral data (Ahrens et al., 2019; Aschwanden and Crosby, 2006; Behroozi et al., 2018; Behroozi and Parnin, 2018; Castelhano et al., 2018; Fakhoury et al., 2020, 2018; Fritz

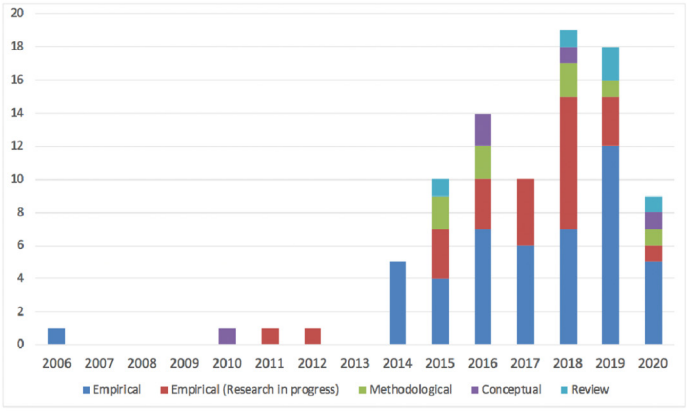
Table 3
Outlets with >2 NeuroSE publications (N=89).

Outlet (short)	Outlet (long)	Number	Paper type
ICSE	International Conference on Software Engineering	11	Conference
EMIP	International Workshop on Eye Movements in Programming	10	Conference
ICPC	International Conference on Program Comprehension	6	Conference
SEmotion	International Workshop on Emotion Awareness in Software Engineering	6	Conference
FSE	Symposium on the Foundations of Software Engineering	5	Conference
EMBC	Int'l Conference of the Engineering in Medicine and Biology Society	3	Conference
ACM CHI	ACM Conference on Human Factors in Computing Systems	2	Conference
ESE	Empirical Software Engineering	2	Journal
ICMSES	International Conference on Mobile Software Engineering and Systems	2	Conference
IST	Information and Software Technology	2	Journal
ISSRE	International Symposium on Software Reliability Engineering	2	Conference
ICAC	International Conference on Augmented Cognition	2	Conference
eLife	eLife	2	Journal

Table 4
Thematic orientation of completed empirical papers (N=47).

ID	Code comprehension	Code inspection	Programming	Change task	Bug fixing	Documenting code	General
Ahonen et al. (2016)			x				
Ahonen et al. (2018)			x				
Ahrens et al. (2019)				x	x		
Aschwanden and Crosby (2006)	x						
Behroozi and Parnin (2018)							x
Behroozi et al. (2018)							x
Castelhano et al. (2018)		x					
Couceiro et al. (2019a)	x						
Couceiro et al. (2019b)	x						
Couceiro et al. (2019c)	x						
Crk and Kluthe (2014)	x						
Crk and Kluthe (2016)	x						
Crk et al. (2016)	x						
Doukakis (2019), Doukakis et al. (2020)			x				
Duraes et al. (2016)		x					
Duraisingam et al. (2017)	x						
Fakhoury et al. (2020, 2018)	x	x					
Floyd et al. (2017)	x	x					
Fritz et al. (2014)	x						
Fucci et al. (2019)	x						
Girardi et al. (2020)				x			
González et al. (2015)			x		x	x	
Huang et al. (2019)	x						
Ikramov et al. (2019)			x				
Ikutani and Uwano (2014)	x						
Ikutani et al. (2020)	x						
Ishida and Uwano (2019a)	x						
Ishida and Uwano (2019b)	x	x					
Ivanova et al. (2020)	x						
Jbara and Feitelson (2015)	x						
Kosti et al. (2018)	x						
Lee et al. (2016)	x						
Lee et al. (2017)	x						
Liu et al. (2020)	x						
Medeiros et al. (2019)	x						
Müller and Fritz (2015)				x			
Müller and Fritz (2016)							x
Nakagawa et al. (2014)	x						
Peitek et al. (2018a)	x						
Siegmund et al. (2014)	x						
Siegmund et al. (2017)	x						
Vrzakova et al. (2020)		x					
Wulff-Jensen et al. (2019)	x						
Yamamoto et al. (2016)			x				
Yeh et al. (2017)	x						
Züger and Fritz (2015)							x
Züger and Fritz (2018)							x
Total	**30**	**6**	**6**	**3**	**1**	**1**	**5**

et al., 2014; Ishida and Uwano, 2019b; Jbara and Feitelson, 2015; Lee et al., 2017; Müller and Fritz, 2015; Vrzakova et al., 2020; Wulff-Jensen et al., 2019; Züger et al., 2018). For example, Züger et al. (2018) combined heart-related measurements with measurements of computer interactions, sleep, and physical activity and Vrzakova et al. (2020) combined skin-related measurements with touch and behavioral eye tracking metrics. An overview of the methods used in the 47 completed empirical papers can be found in Appendix D. To provide guidance in choosing suitable methods for their own investigations, both in laboratory and field settings, we refer the interested reader to Riedl and Léger (2016). In this publication, both strengths and weaknesses of methods are



	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	Total Papers / Contribution Type
Empirical	1	0	0	0	0	0	0	0	5	4	7	6	7	12	5	**47**
Empirical (Research in progress)	0	0	0	0	0	1	1	0	0	3	3	4	8	3	1	**24**
Methodological	0	0	0	0	0	0	0	0	0	2	2	0	2	1	1	**8**
Conceptual	0	0	0	0	1	0	0	0	0	0	2	0	1	0	1	**5**
Review	0	0	0	0	0	0	0	0	0	1	0	0	1	2	1	**5**
Total Papers / Year	**1**	**0**	**0**	**0**	**1**	**1**	**1**	**0**	**5**	**10**	**14**	**10**	**19**	**18**	**9**	**89**

Fig. 3. Development of NeuroSE Research Over Time (N=89).

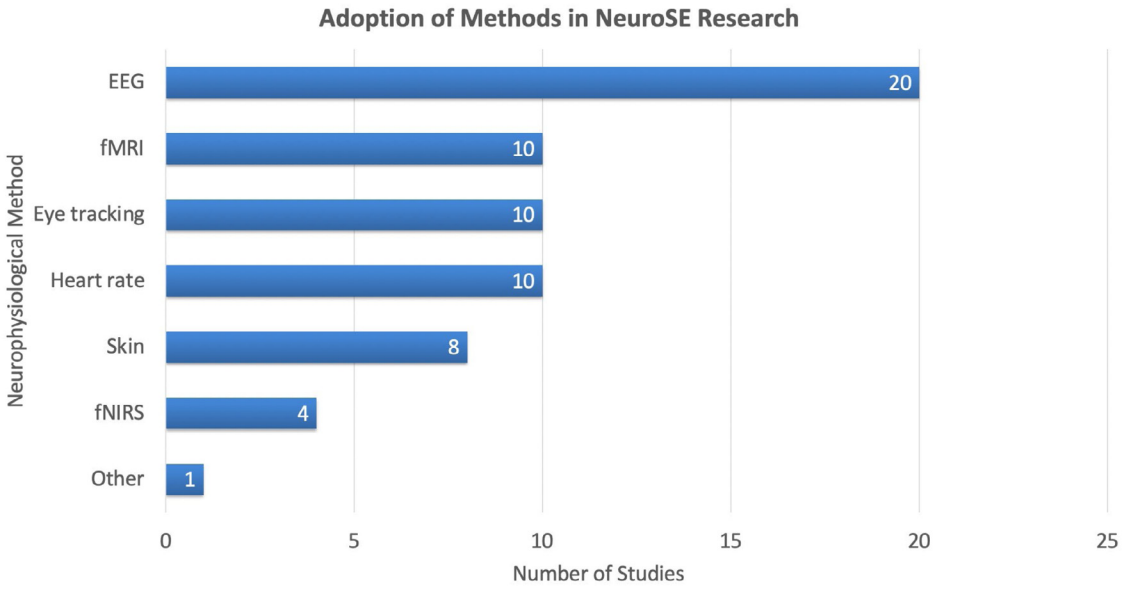


Fig. 4. Adoption Rate of Methods in Completed Empirical NeuroSE Research (N=47).

indicated. A more detailed discussion of methods can be found in Senior et al. (2009).

4.3.2. Overview of collected neurophysiological measures

An overview of the collected neurophysiological measures categorized by method is provided in Tables 5 and 6. An overview of the used measurement instruments categorized by method is provided in Appendix E.

EEG measures. Most studies using EEG focused on the analysis of *frequency bands* (i.e., the EEG signal was decomposed into frequency bands: Alpha, Beta, Gamma, Delta, and Theta, which were then analyzed for power differences). In addition, several studies considered *frequency band ratios*, e.g., the ratio of Alpha/Beta. In addition, one study used simple *time-domain features* like (normalized) signal average and variance. A few studies considered *compound signals* (e.g., attention) provided by the used EEG device through pre-built algorithms. Moreover, 3 studies used *eye blinking rate* as a feature, i.e., the number of eye blinks per minute. Two studies captured *interhemispheric differences* (i.e., power differences between right and left hemisphere) and two studies considered non-directed *functional connectivity measures* (i.e., statistical associations between spatially distinct brain areas). More specifically, one study used cross correlations of frequency band power between electrodes and one study used Phase Locking Values, i.e., measuring phase synchrony between pairs of electrodes. Our analysis also shows that apart from Ikramov et al. (2019), Lee et al. (2017, 2016), Medeiros et al. (2019) who used research-grade EEG instruments by the company Brain Products (2 studies), Compumedics (1 study) and Mitsar (1 study), all remaining EEG studies used low-cost consumer-grade EEG devices for data collection. Emotiv EPOC was used 5 times, NeuroSky mindset headset 3 times, NeXus 10 MARK II 3 times, and NeuroSky mindwave headset, Emotiv EPOC+, BrainLink Pro, and BIOPAC MP150 were used once (for a discussion of potential

Table 5
Overview of collected neurophysiological measures (Part 1, brain activity measurement tools).

Method	Measures	Studies
EEG	Frequency bands	(Crk et al., 2016; Crk and Kluthe, 2016, 2014; Doukakis, 2019; Doukakis et al., 2020; Duraisingam et al., 2017; Fucci et al., 2019; Girardi et al., 2020; González et al., 2015; Ikramov et al., 2019; Ishida and Uwano, 2019b,a; Kosti et al., 2018; Lee et al., 2016; Medeiros et al., 2019; Müller and Fritz, 2015; Yamamoto et al., 2016; Yeh et al., 2017; Züger and Fritz, 2015)
	Frequency band ratios	(Duraisingam et al., 2017; Fritz et al., 2014; Fucci et al., 2019; Girardi et al., 2020; González et al., 2015; Müller and Fritz, 2015; Yamamoto et al., 2016; Züger and Fritz, 2015)
	Compound signals	(Fritz et al., 2014; Fucci et al., 2019; Girardi et al., 2020; González et al., 2015; Müller and Fritz, 2015; Züger and Fritz, 2015)
	Time domain features	(Medeiros et al., 2019)
	Eye blink rate	(Fritz et al., 2014; Müller and Fritz, 2015; Züger and Fritz, 2015)[a]
	Interhemispheric differences	(Duraisingam et al., 2017; Lee et al., 2016)
	Functional connectivity measures	(Kosti et al., 2018; Lee et al., 2016)
fMRI	BOLD contrast	(Castelhano et al., 2018; Duraes et al., 2016; Floyd et al., 2017; Huang et al., 2019; Ikutani et al., 2020; Ivanova et al., 2020; Liu et al., 2020; Peitek et al., 2018a; Siegmund et al., 2017, 2014)
	Functional connectivity measures	(Castelhano et al., 2018)
fNIRS	Oxy-Hb	(Ikutani and Uwano, 2014; Nakagawa et al., 2014)
	Oxy-Hb/DeOxy-Hb	(Fakhoury et al., 2020, 2018; Huang et al., 2019)

[a]Please note that eye blinks are not derived from the EEG signal and therefore do not constitute a real EEG measure. They are derived from the electrooculogram (EOG) signal collected by the EEG device and are typically used for artifact removal. We list eye blinks here since they were collected using EEG devices.

limitations of using consumer-grade EEG devices in a research context, please see Section 5.3).[3]

fMRI measures. All fMRI studies explicitly stated the captured signal, the *BOLD contrast*. In addition, one study used directed *functional connectivity measures*, i.e., they exploited temporal precedence information to detect the influence of brain regions and its direction (Kim et al., 2013). 7 out of 10 studies were run in a 3T Magnetom Trio Tim MRI scanner by Siemens. The three remaining studies were conducted in a 3T Magnetom Prisma scanner by Siemens, a 3T General Electric MR750, and a 3T Philips Achieva Multix X-Series scanner.

fNIRS measures. 2 studies used the concentration of oxygenated hemoglobin in the cerebral blood flow (i.e., *Oxy-Hb*), while 2 studies used the ratio of oxygenated and deoxygenated hemoglobin (i.e., *Oxy-Hb/DeOxy-Hb*). Devices for data collection included a NeXus10 with a Nexus HEG sensor, a Wearable Hikari Topography WOT-200, a fNIR100 by BIOPAC, and CW6 a fNIRS by TechEn Inc.

Eye-related measures. The studies covered by our review analyzed *pupil size*, i.e., a measurement describing how large the pupil is, *LH ratio*, i.e., the low frequency/high frequency ratio of the pupil size variability (Shaffer and Ginsberg, 2017), *eye blink rate*, i.e., the number of eye blinks over time (Holmqvist and Andersson, 2017), and *eye blink duration*, i.e., the average duration of the eye blinks (Holmqvist and Andersson, 2017).[4] Devices included a low-cost eye tracker by Eye Tribe 60 Hz (3 studies), SMI eye tracking glasses 60 Hz (2 studies), a SMI eye tracker (2 studies),[5] a Tobii TX 300 (1 study), a Tobii X3-120 (1 study), and an ASL eye tracking system (1 study).

Heart-related measures. Our results show that studies used ECG-measures, *heart rate (HR)*, i.e., the number of heart beats per minute (7 studies) and *heart rate variability (HRV)*, i.e., the changes in the time intervals between consecutive heartbeats called interbeat intervals (IBIs) (Shaffer and Ginsberg, 2017) (8 studies). More specifically, we could observe the usage of several HRV time-domain measures like SDNN (i.e., standard deviation of NN intervals[6]), RMSSD (i.e., root mean square of successive RR interval differences), pNN20 and pNN50 (i.e., percentage of successive RR intervals[7] that differ by more than 20 ms and 50 ms respectively). In addition, two studies used frequency domain features like the low/high frequency ratio of the ECG RR interval variability (Shaffer and Ginsberg, 2017). Moreover, our results show the usage of *blood volume pulse (BVP)* measurement (4 studies). We identified use of wearable research-grade devices like Empatica E4 (PPG signal, 3 studies), Empatica E3 (PPG signal, 2 studies), BiosignalsPlus toolkit (ECG sensor, 2 studies), and eMotion Faros 180° (ECG sensor, 2 studies) as well as consumer-grade devices like SenseCore Chest Strap,[8] Polar H7 chest belt (ECG signal, 1 study), and Fitbit Charge 2 (ECG sensor, 1 study).

Skin-related measures. In terms of skin-related measures, the analyzed studies used the EDA signal and considered measures

[3] BIOPAC MP150 and NEXUS 10 MARK II can be used not only to measure EEG, but can also be equipped with sensors to measure other physiological indicators such as HR. The authors of the papers that used these devices did, however, not specify the exact types of sensors that were used aside from their intention to measure EEG

[4] Please note that fixations and saccades are not included here, since they are *not* predominantly reflexive; cf. Exclusion Criterion 4.

[5] No information concerning the exact SMI model was provided. Currently available screen-based eye tracker models from SMI include the SMI Red tracker with sampling frequencies of 60 Hz and 120 Hz.

[6] NN intervals are interbeat intervals from which artifacts have been removed.

[7] RR intervals are interbeat intervals between successive heartbeats.

[8] This sensor is not available anymore.

Table 6
Overview of collected neurophysiological measures (Part 2, autonomic nervous system measurement tools).

Method	Measures	Studies
EYE	Pupil size	(Ahrens et al., 2019; Aschwanden and Crosby, 2006; Behroozi and Parnin, 2018; Fritz et al., 2014; Jbara and Feitelson, 2015; Müller and Fritz, 2015; Wulff-Jensen et al., 2019)
	LH ratio of pupil size	(Couceiro et al., 2019a,b)
	Eye blink rate	(Aschwanden and Crosby, 2006; Behroozi et al., 2018)
	Eye blink duration	(Behroozi et al., 2018)
SKIN	EDA signal: Skin Conductance Level and Skin Conductance Response	(Ahonen et al., 2018; Fritz et al., 2014; Fucci et al., 2019; Girardi et al., 2020; Müller and Fritz, 2016, 2015; Vrzakova et al., 2020; Züger and Fritz, 2015)
	Temperature sensor: Skin temperature	(Müller and Fritz, 2016, 2015; Züger and Fritz, 2015)
HEART	Blood volume pulse (BVP)	(Fucci et al., 2019; Girardi et al., 2020; Müller and Fritz, 2015; Züger and Fritz, 2015)
	Heart rate (HR)	(Ahonen et al., 2018, 2016; Fucci et al., 2019; Girardi et al., 2020; Müller and Fritz, 2016, 2015; Züger et al., 2018; Züger and Fritz, 2015)
	Heart rate variability (HRV)	
	• Time-domain features	(Ahonen et al., 2018, 2016; Couceiro et al., 2019c; Fucci et al., 2019; Müller and Fritz, 2016, 2015; Züger et al., 2018; Züger and Fritz, 2015)
	• SDNN	(Ahonen et al., 2018, 2016; Fucci et al., 2019; Züger et al., 2018; Züger and Fritz, 2015)
	• RMSSD	(Ahonen et al., 2016; Fucci et al., 2019; Müller and Fritz, 2016; Züger et al., 2018)
	• pNN20	(Züger and Fritz, 2015)
	• pNN50	(Müller and Fritz, 2016; Züger et al., 2018; Züger and Fritz, 2015)
	• Frequency domain features	(Couceiro et al., 2019a,c)
	• LH ratio	(Couceiro et al., 2019a)
Other	Respiratory rate	(Müller and Fritz, 2016)

related to the *skin conductance level* (SCL, tonic component) and the *skin conductance response* (SCR, phasic component). The tonic component refers to changes in skin conductance which are not causally related to an eliciting stimulus, while the phasic component refers to conductance changes that occur within a discrete time window (often 1–3 s) after stimulus perception (Naqvi and Bechara, 2006). In addition, several studies used a temperature sensor to measure *skin temperature* (note that skin temperature increases as a consequence of activation of the sympathetic part of the ANS). In all cases, wearable, research-grade devices were used for data collection. Specifically, we found that Empatica E4 was used in 3 studies, Empatica E3 in 2 studies, a Shimmer3 GSR+ device in 2 studies, and an Affective Q sensor in 1 study.

Other measures. 1 study considered the *respiratory rate*, i.e., the number of breaths taken per minute and used the chest strap SenseCore[9] for data collection.

4.3.3. Usage of methods

To guide the interested reader to the actual usage of the different neurophysiological methods, Appendix G provides an overview of the research questions answered using the different methods. In the remainder of this section, we briefly summarize the type of research questions addressed, structured by themes.

This overview shows that fMRI studies primarily looked into which brain regions are activated during different software development activities (e.g., code comprehension) or task-related events (e.g., "bug suspicion"). For example, Siegmund et al. (2014) investigated the question "Which brain regions are activated during program comprehension?" or Duraes et al. (2016) looked into the question "What are the brain activation patterns associated with bug confirmation?". Moreover, several fMRI studies investigated differences in brain activation and brain regions involved between different experimental conditions. For example, Floyd et al. (2017) examined the question "Are neural representations of programming languages and natural languages distinct?" and Liu et al. (2020) investigated "Is the neural signature of code comprehension similar to other culturally derived symbol systems (i.e., logic and math) or similar to natural language?". Differences in brain activation between different conditions were also in the focus of several EEG studies. For example, Kosti et al. (2018) investigated "How do comprehension and syntax tasks differ in terms of patterns of brain activation?".

Another group of studies looked for correlates between brain activity and constructs such as cognitive load, task difficulty, or task performance. For example, using fMRI Huang et al. (2019) investigated "What is the impact of task difficulty on brain activation?". As another example, Duraisingam et al. (2017) asked "Is task difficulty reflected in EEG electrical signal within programming comprehension tasks?" or based on fMRI Castelhano et al. (2018) examined "Does activation in the anterior insula correlate with bug detection precision?". Note that the insula is a brain region which is related to emotionally aversive stimuli, including spiders and snakes, anticipation of physical pain, excessive prices in purchase decisions, as well as situations characterized by uncertainty, ambiguity, and distrust (for a collection of references, see a brief review by Riedl et al. (2010b, p. 405).

Various EEG studies and studies using (multi-modal) measurements related to ANS activity looked into the efficacy of neurophysiological measures to predict different dependent variables (e.g., cognitive load, interruptibility, affective state) in both offline and online settings. For example, Couceiro et al. (2019b) examined "Can a developer's cognitive load be measured during code reading using pupillography?" and Fritz et al. (2014) addressed the question "Can we acquire psycho-physiological measures from eye tracking, EDA and EEG sensors to accurately predict task difficulty?". Similarly, Züger et al. (2018) investigated the question "Can we build a classifier that predicts a software developer's interruptibility accurately in the field?". Moreover, several studies investigated which combination of neurophysiological measures is best suited for a certain prediction and aimed to shed light on how they compare to more traditional measures. For example, Fritz et al. (2014) addressed the question "Which combination of sensor and associated features works best?" and

[9] This sensor is not available anymore.

Müller and Fritz (2016) posed the question "How do biometrics compare to more traditional metrics for predicting perceived difficulty and detecting quality concerns?".

Eye tracking was used together with other modalities for the fine-grained analysis of cognitive load. For example, Couceiro et al. (2019a) investigated "Can eye tracking together with HRV and pupillography be used to identify non intrusively code lines (and even lexical tokens inside code lines) that correspond to mental effort peaks?". Similarly, Fakhoury et al. (2020, 2018) posed the question "Can developers' cognitive load be accurately associated with identifiers' terms using fNIRS and eye tracking devices?". In addition, several studies using eye tracking investigated pupil size and/or eye blinks along with behavioral eye tracking metrics (e.g., fixations and/or saccades) (Ahrens et al., 2019; Aschwanden and Crosby, 2006; Jbara and Feitelson, 2015; Wulff-Jensen et al., 2019). For example, Aschwanden and Crosby (2006) investigated "What are the scanning patterns during program comprehension?", while Jbara and Feitelson (2015) looked into "Are developers' visual efforts equally divided among regular segments?". Based on the formulation of these questions it is obvious that the eye measures related to ANS activity (pupil size, eye blink) did not play a major role in the reviewed studies. Rather, studies were focused on fixation patterns. We nevertheless kept these studies in our review, but only analyzed those aspects of the papers that were related to ANS activity.

4.4. How was the empirical NeuroSE research conducted (RQ4)?

In this section we report additional details on the completed empirical studies (N=47). This includes details concerning study participants, stimuli, experimental design, independent and dependent variables as well as data analysis approaches used.

4.4.1. Study participants, stimuli, and experimental design

Study Participants. Sample sizes of the empirical studies ranged from 2 to 70 participants with a median of 17. The distribution of sample sizes is depicted in Fig. 5.

Female participation ranged from 0% to 63.16% with a median of 13.39%. Participants' age range was from 16 to 60 years with a median age of 26 years (note that information on age was not provided in 20 out of 47 studies). When analyzing sample sizes per method we found the following median sample sizes: fMRI (18, 10 studies), fNIRS (13, 4 studies), EEG (10, 20 studies), eye tracking (18.5, 10 studies), heart-related measurements (22, 10 studies), and skin-related measurements (16, 8 studies). The majority of studies (27) relied on student subjects. Two studies used faculty members in addition to students (i.e., Jbara and Feitelson (2015), Kosti et al. (2018)). Five studies used both students and professional software developers (i.e., Ahrens et al. (2019), Lee et al. (2017), Müller and Fritz (2015), Siegmund et al. (2017), Züger and Fritz (2015)). 6 studies solely relied on professional software developers (i.e., Castelhano et al. (2018), Duraes et al. (2016), Fritz et al. (2014), Müller and Fritz (2016), Vrzakova et al. (2020), Züger et al. (2018)). 7 studies did not further specify whether their participants were students or professional developers. Appendix I summarizes details concerning the study participants.

Stimuli. Most studies used simple code snippets between 3 and 60 lines. In turn, a few studies used realistic tasks like adding a feature to an existing library (i.e., Ahrens et al. (2019), Müller and Fritz (2015), Züger and Fritz (2015)) and 3 studies were even conducted in a real-world setting (i.e., Müller and Fritz (2016), Vrzakova et al. (2020), Züger et al. (2018)). Programming languages included primarily Java (28 out of 47), C (7 out of 47), Python (3 out of 47), Scratch (2 out of 47), but also C/C++, C#, and Processing (1 in each case). Detailed information about the tasks and their complexity is not provided in all studies. Appendix J provides an overview of task characteristics, programming languages, and the size of the used code snippets.

Experimental Design. With the exception of Ahonen et al. (2018, 2016), Müller and Fritz (2016), Vrzakova et al. (2020), Züger and Fritz (2015), Züger et al. (2018), all studies were conducted in the lab. The study presented in Züger and Fritz (2015) reports on the results of two studies (one in the lab and one in the field). Most of the studies (namely 37) used a within-subject design and conducted repeated measurements of the same condition at different points in time (e.g., by conducting multiple trials) and/or applied different experimental conditions. In addition, 3 studies used a between-subject design, two studies used a mixed design, and 5 studies did not apply an experimental design.

4.4.2. Dependent and independent variables

The majority of studies used neurophysiological data as dependent variable. An overview of independent variables whose effect on neurophysiological data was investigated is shown in Table 7. The independent variables are categorized into task-specific factors, developer-specific factors, team-specific factors, task performance, and context factors. Task-specific factors are further sub-divided into software development activities, task-related events, and task characteristics. These categories emerged as a result of a bottom-up coding process of the literature.

We found a substantial number of studies that looked into brain activation depending on different *task-specific factors*. In particular, several studies looked into brain activation patterns for specific *software development activities* (e.g., code comprehension, code inspection). For example, Peitek et al. (2018a) and Siegmund et al. (2017, 2014) investigated which areas of the brain become activated while engaging in code comprehension tasks in contrast to syntax tasks.[10] Moreover, differences in brain activation patterns as well as brain connectivity patterns during code comprehension vs. syntax tasks are investigated in Kosti et al. (2018). Brain activation during code comprehension versus code inspection is examined in Castelhano et al. (2018); specifically, this study contrasted code inspection (i.e., searching for bugs) with code understanding (i.e., reading neutral code). Programming, bug fixing, and documenting code is compared in González et al. (2015). Moreover, several studies looked into the difference between code and prose. For example, Castelhano et al. (2018) contrasted brain activation during source code understanding and pseudo-code text reading. In addition, Floyd et al. (2017) compared code inspection with prose review. Several recent studies examined differences in brain activation during code related activities (code comprehension or data structure manipulations), as well as brain activation in regions related to mental rotation (Huang et al., 2019), math/logic/language/multi-source interference (Liu et al., 2020), sentence comprehension, nonword reading, and hard memory tasks (Ivanova et al., 2020).

Existing studies not only contrasted different software development activities, but also different *task-related events*. For example, based on fMRI Duraes et al. (2016) and Castelhano et al. (2018) analyzed how brain activation differs for the events "bug suspicion" and "bug confirmation". Moreover, several studies investigated the impact of different *task characteristics* on neurophysiological measures. For example, the study described in Siegmund et al. (2017) investigated the role of bottom-up program comprehension and comprehension with semantic cues

[10] Syntax tasks require subjects to spot syntactical errors without requiring them to understand the behavior of the program.

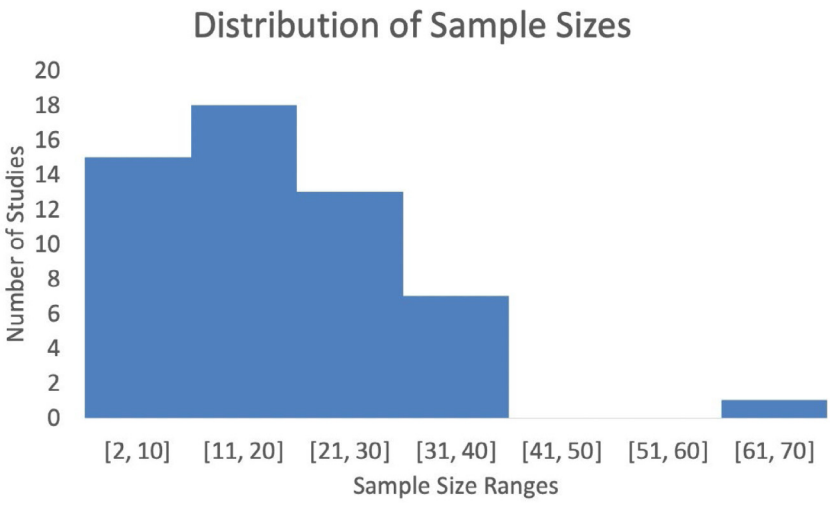


Fig. 5. Distribution of Sample Sizes.

Table 7
Categorization of independent variables.

Overview of Independent Variables (with neurophysiological data as dependent variables)		
Task-specific factors	Software development activities	code comprehension vs. syntax (Kosti et al., 2018; Peitek et al., 2018a; Siegmund et al., 2017, 2014) code comprehension vs. fake code/math/logic/language/multi-source interference (Liu et al., 2020) code vs. sentence comprehension vs. nonword reading (Ivanova et al., 2020) code comprehension vs. hard working memory task (Ivanova et al., 2020) code understanding vs. pseudocode reading (Castelhano et al., 2018) code comprehension vs. inspection (Castelhano et al., 2018) programming vs. bug fixing vs. documenting (González et al., 2015) data structure manipulation vs. mental rotation (Huang et al., 2019)
	Task-related events	bug suspicion vs. bug confirmation (Castelhano et al., 2018; Duraes et al., 2016)
	Task characteristics	task difficulty (Couceiro et al., 2019a,b; Duraisingam et al., 2017; Huang et al., 2019; Ikutani and Uwano, 2014; Nakagawa et al., 2014; Yeh et al., 2017) problem type (Ikutani and Uwano, 2014) bottom-up versus semantic cues (Siegmund et al., 2017) presence of English identifier (Ivanova et al., 2020) layout (Siegmund et al., 2017) linguistic antipatterns and structural inconsistencies (Fakhoury et al., 2020, 2018) structural and textual features (Wulff-Jensen et al., 2019) code regularity (Jbara and Feitelson, 2015) attention data representation (Ahrens et al., 2019) textual vs. visual programming language (Doukakis, 2019; Doukakis et al., 2020) paper versus whiteboard (Behroozi et al., 2018)
Developer-specific factors		expertise (Crk et al., 2016; Lee et al., 2016)
Team-specific factors		pair programming (collaborating dyads vs. shuffled pairs) (Ahonen et al., 2018, 2016) programming role solo, pair/navigator, pair/driver (Ikramov et al., 2019) role differences (driver, navigator) in task-relevant events (running and testing code) (Ahonen et al., 2018)
Task performance		finding implementation strategy (Yamamoto et al., 2016) correctness of responses (Crk et al., 2016) passing and failing in task-relevant events running and testing code (Ahonen et al., 2018) completion of comprehension step (Ishida and Uwano, 2019b,a) completion of judgment step (Ishida and Uwano, 2019a)
Context factors		listening to music (Ikramov et al., 2019)

in terms of brain activation and examined how layout and beacons in source code influence program comprehension. In addition, Fakhoury et al. (2018) investigated the impact of linguistic antipatterns and structural inconsistencies on perceived task difficulty and brain activation. The role of task difficulty on brain activation is investigated, in turn, in Couceiro et al. (2019a,b), Duraisingam et al. (2017), Huang et al. (2019), Ikutani

and Uwano (2014), Nakagawa et al. (2014), Yeh et al. (2017). Additionally, studies looked into *developer-specific factors* like expertise (i.e., Crk et al. (2016), Lee et al. (2016)). For example, Lee et al. (2016) investigated the impact of expertise on brain activation. Several studies picked up *team-specific factors*. For example, Ahonen et al. (2018, 2016) investigated synchrony in heart- and skin-related signals during pair programming. A few studies looked into the relationship between *task performance* and brain activation (i.e., Crk et al. (2016), Yamamoto et al. (2016)). For example, Crk et al. (2016) examined the relationship of expertise and task performance (in terms of correct answers) with brain activation patterns. Finally, *context factors* were addressed in one study; Ikramov et al. (2019) investigated the role of music on brain activation during programming.

Fig. 6 shows a Sankey diagram summarizing how the studies in which neurophysiological data was used as dependent variable were conducted. The studies by Ahonen et al. (2018, 2016), Ahrens et al. (2019), Behroozi et al. (2018), Castelhano et al. (2018), Couceiro et al. (2019b), Crk et al. (2016), Doukakis (2019), Doukakis et al. (2020), Duraes et al. (2016), Duraisingam et al. (2017), Fakhoury et al. (2020, 2018), González et al. (2015), Huang et al. (2019), Ikramov et al. (2019), Ikutani and Uwano (2014), Ishida and Uwano (2019b,a), Ivanova et al. (2020), Jbara and Feitelson (2015), Kosti et al. (2018), Lee et al. (2016), Liu et al. (2020), Nakagawa et al. (2014), Peitek et al. (2018a), Siegmund et al. (2017, 2014), Wulff-Jensen et al. (2019), Yamamoto et al. (2016), Yeh et al. (2017) formed the basis for Fig. 6. Sankey diagrams can be used to display flows from one set of values to another along with their quantities. The Sankey diagram depicted in Fig. 6 connects the software development activities (column 1), the category of the independent variables (column 2), the independent variables (column 3), the neurophysiological measures (column 4), and the neurophysiological methods (column 5). The thickness of the links provides an indication of how often a certain connection was observed. Colors were chosen randomly and hence do not have a specific meaning. For example, when focusing on column 1 the Sankey diagram shows that the majority of studies focused on code comprehension followed by programming. When looking at column 2 the figure shows that a strong focus was on task-specific factors. The links between column 1 and column 2 show, for example, that task-specific factors were investigated in the context of various software development activities. Looking at connections between column 2 and column 3 it becomes apparent that studies put a strong focus on task characteristics followed by software development activities. The links between column 3 and column 4 show that differences between software development activities were mainly investigated using measures obtained from fMRI, EEG, and fNIRS. The substantially higher number of outgoing links when compared to incoming links in column 3 shows that several studies used different measures in combination. The figure also shows that eye tracking was mainly used to investigate different task characteristics. Moreover, it is clearly visible that methods related to brain activation (i.e., fMRI, EEG, and fNIRS) were much more frequently used if compared to methods related to ANS activity.

In addition to the studies in which neurophysiological data were used as dependent variable, several studies also used neurophysiological measures as independent variables. An overview of dependent factors that were predicted based on neurophysiological data is shown in Table 8. The dependent variables are categorized into task-specific factors, developer-specific factors, mental states, and task performance. These categories emerged as a result of a bottom-up coding process of the literature.

All of these studies followed a data-driven approach and extracted various features from the neurophysiological data (in several cases along with additional data sources like, for example, behavioral data). Neurophysiological data was used to predict *task-specific* factors like the software development activity (e.g., Floyd et al. (2017)), the task (sub)category (Ikutani et al., 2020), the problem structure (Liu et al., 2020), or the presence of quality concerns (Müller and Fritz, 2016). For example, Floyd et al. (2017) distinguished code comprehension, code inspection, and prose review. Moreover, neurophysiological measures were used to predict *developer-specific* factors like expertise (Crk and Kluthe, 2014; Lee et al., 2017) or *task performance* (e.g., correctness of responses Crk and Kluthe, 2016). In addition, several studies used neurophysiological data to predict different psychological constructs related to cognitive or emotional processes of the developer. In the remainder of this paper, we refer to these constructs as *mental states* (both cognitive and affective). Drawing upon Cowley et al. (2016), we define a mental state as any interesting aspect of an individual's state that can be interpreted from this individual's physiology and thus measured by sensors. Examples of mental states include cognitive load, affective state, perceived progress, interruptibility, and stress. For example, Fritz et al. (2014) used EEG, eye tracking, and EDA features to predict cognitive load during program comprehension. Similarly, Duraisingam et al. (2017) investigated whether the difficulty of a program comprehension task and the associated cognitive load can be predicted from EEG. In addition, Müller and Fritz (2015) used EEG, eye tracking, skin-related measures, and heart-related measures to predict a developer's affective state and perceived progress. The study by Züger and Fritz (2015) predicted interruptibility from EEG, skin-related measures, and heart-related measures. Similarly, Züger et al. (2018) considered heart-related data along with computer interaction, sleep-, and physical activity-related data to predict interruptibility.

Appendix F provides a detailed overview of all the measures grouped by neurophysiological method that were used to predict the various dependent variables.

Fig. 7 shows a Sankey diagram providing an overview of all the studies in which neurophysiological data was used as independent variable to predict an outcome variable (i.e., it is based on the studies by Behroozi and Parnin (2018), Couceiro et al. (2019c), Crk and Kluthe (2014), Duraisingam et al. (2017), Floyd et al. (2017), Fritz et al. (2014), Fucci et al. (2019), Girardi et al. (2020), Ikutani et al. (2020), Kosti et al. (2018), Lee et al. (2017), Liu et al. (2020), Medeiros et al. (2019), Müller and Fritz (2015, 2016), Vrzakova et al. (2020), Züger and Fritz (2015), Züger et al. (2018)). The Sankey diagram connects the software development activities (column 1), the category of the dependent variables (column 2), the dependent variables (column 3), and the neurophysiological methods that were used to predict the dependent variables (column 4). Looking at column 1, we see that code comprehension was the focus of the majority of studies, followed by studies on general software development activities. When looking at column 2 we see that mental states were the prevalent category of dependent variables. The links between column 1 and column 2, for example, show that dependent variables of all four categories were predicted during code comprehension, while for change tasks only mental states were predicted. For the node "Mental State" the figure shows one more outgoing link than incoming links. This indicates that one of the studies predicted two distinct mental states. In turn, for the node "Task-specific factor" there is a higher number of incoming than outgoing links, which signifies that one study investigated more than one software development activity. Focusing on the connections between column 2 and column 3 we can observe that cognitive load was by far the most popular dependent variable that was predicted using neurophysiological measurements. The high number of links between column 3 and column 4 highlights that the same neurophysiological methods were used to predict several dependent variables. For example, for predicting cognitive

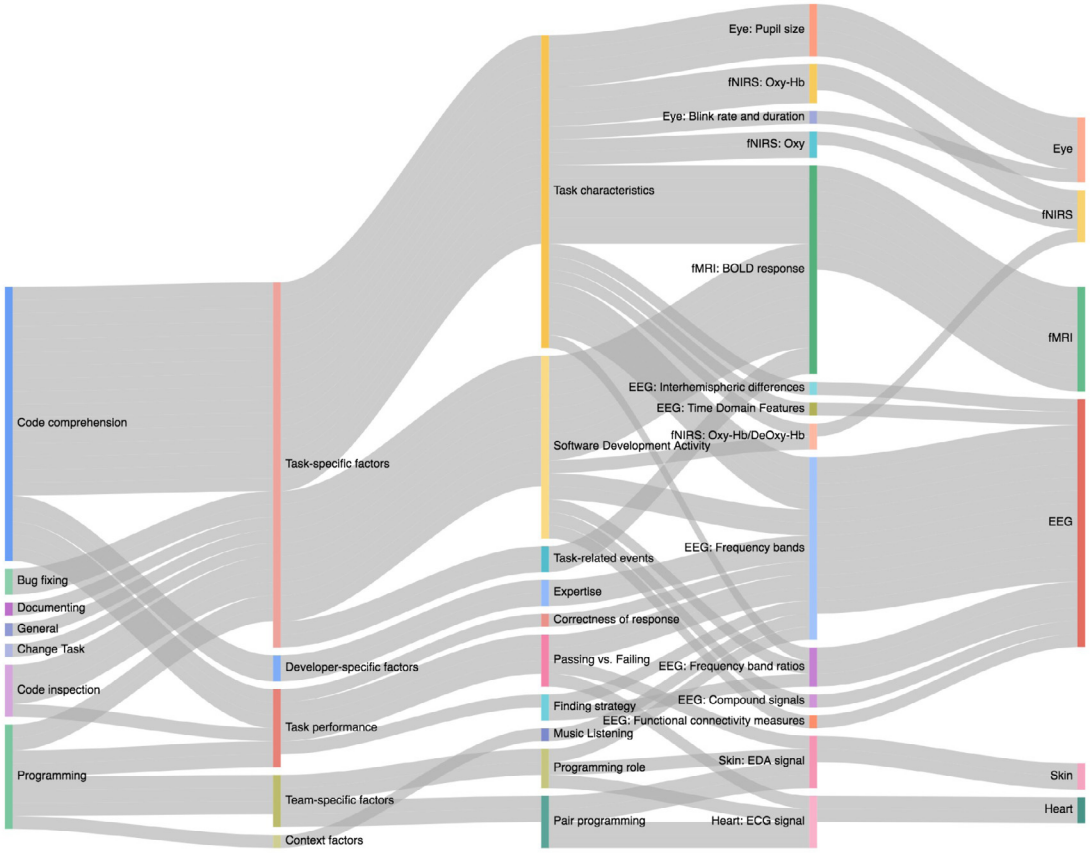


Fig. 6. Sankey diagram of studies in which neurophysiological data was used as dependent variable.

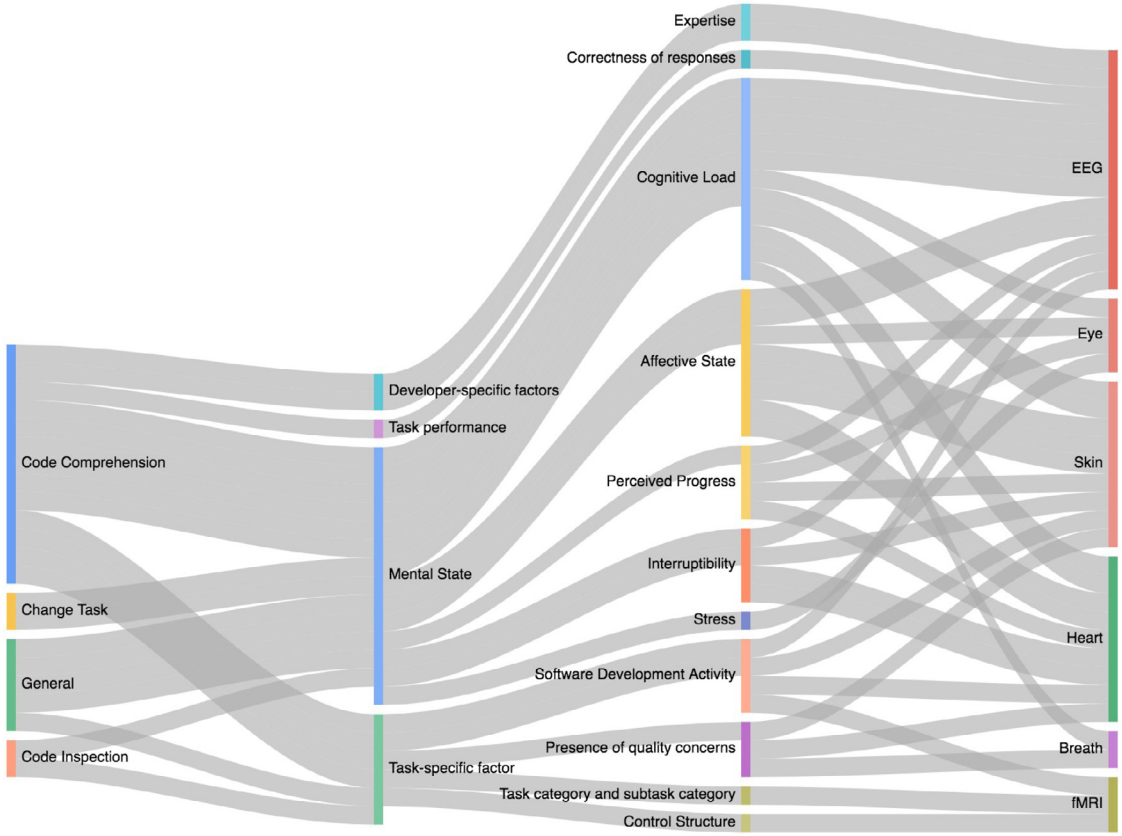


Fig. 7. Sankey diagram of studies where neurophysiological data was used as independent variable.

Table 8
Categorization of dependent variables.

Overview of dependent variables (with neurophysiological measures as independent variables)	
Task-specific factors	Software development activity code comprehension vs. code inspection vs. prose review (Floyd et al., 2017) code comprehension vs. prose comprehension (Fucci et al., 2019) Task category, Task subcategory (Ikutani et al., 2020) Problem structure (if and for statements) (Liu et al., 2020) Presence of quality concerns (Müller and Fritz, 2016)
Developer-specific factors	Expertise (Crk and Kluthe, 2014; Lee et al., 2017)
Mental states	Cognitive Load (Couceiro et al., 2019c; Duraisingam et al., 2017; Fritz et al., 2014; Kosti et al., 2018; Lee et al., 2017; Medeiros et al., 2019; Müller and Fritz, 2016) Interruptibility (Züger and Fritz, 2018, 2015) Affective state (Girardi et al., 2020; Müller and Fritz, 2015; Vrzakova et al., 2020) Perceived progress (Müller and Fritz, 2015) Stress (Behroozi and Parnin, 2018)
Task performance	Correctness of responses (Crk and Kluthe, 2016)

load EEG, eye tracking as well as heart, skin, and breath-related measures were used. Moreover, the higher number of outgoing links when compared to incoming links at column 3 signifies that several studies relied on multi-modal measurements combining several neurophysiological methods. For example, to predict the presence of quality concerns skin, heart, and breath-related measures were used. If compared to Fig. 6, it can be seen that methods related to ANS activity appeared much more frequently and were more popular when compared to methods related to the CNS (i.e., fMRI, EEG, and fNIRS). Thus, our review reveals that brain activity measurement is predominantly used as dependent variable (predicted *by* other factors), while ANS activity measurement is frequently used as independent variable (*to* predict other factors).

For each of the 47 completed empirical studies Appendix K provides a detailed overview of the research questions along with the independent and dependent variables that were used to answer these questions and a concise summary of the main findings (cf. supplementary material). Whenever neurophysiological data was used to operationalize these variables the used measures are listed.

4.4.3. Data analysis

Data analysis methods applied to examine research questions where neurophysiological data was the dependent variable include descriptive statistics and visual analysis, the testing for differences in means as well as the testing of relationships (i.e., regression and correlation), and brain connectivity analysis.

- Descriptive statistics and visual analysis: Several studies only provided descriptive statistics and/or performed a visual analysis regarding neurophysiological data (i.e., Couceiro et al. (2019a,b), Doukakis (2019), Doukakis et al. (2020), Ishida and Uwano (2019b), Jbara and Feitelson (2015), Nakagawa et al. (2014)).
- Testing for differences: Multiple studies applied statistics to test for differences in means, i.e., Ryan method (Ikutani and Uwano (2014)), ANOVA (González et al., 2015; Lee et al., 2016) t-test (Ahrens et al., 2019; Huang et al., 2019; Ishida and Uwano, 2019a; Jbara and Feitelson, 2015; Yamamoto et al., 2016), ANCOVA (Crk et al., 2016), paired t-test and ANOVA (Wulff-Jensen et al., 2019; Yeh et al., 2017), Mann–Whitney U test (Fakhoury et al., 2020; Ikramov et al., 2019), Wilcoxon Signed-Rank Test (Behroozi et al., 2018; Fakhoury et al., 2018), Kruskal–Wallis (Duraisingam et al., 2017), a permutation test (Ahonen et al., 2018, 2016). Moreover, the study described in Fakhoury et al. (2018) used the Simple Matching Coefficient to compare the similarity and diversity of sample sets. In addition, Ahonen et al. (2018) used the Minimum-width envelope method to compare physiological signals and identify periods where the different conditions significantly differed.
- Testing relationships: Multiple studies performed correlation analysis to answer (some of) their research questions (i.e., Aschwanden and Crosby (2006), Castelhano et al. (2018), Fakhoury et al. (2018), Floyd et al. (2017), Fucci et al. (2019), Ikramov et al. (2019), Ikutani et al. (2020), Kosti et al. (2018), Lee et al. (2016), Peitek et al. (2018a), Yeh et al. (2017)). Moreover, regression analyses were used by Girardi et al. (2020), Müller and Fritz (2015), Vrzakova et al. (2020). In addition, to analyze the BOLD response in brain imaging studies, 5 fMRI studies (Castelhano et al., 2018; Duraes et al., 2016; Peitek et al., 2018a; Siegmund et al., 2017, 2014) used a Random Effects General Linear Model (a type of regression analysis) to determine how brain activation changes across the different experimental conditions. Moreover, 3 studies (Huang et al., 2019; Ivanova et al., 2020; Liu et al., 2020) used a multi-level approach for analyzing the BOLD response. In a first step they estimated parameters at an individual level for each subject using a General Linear Model (GLM) and then conducted a group-level analysis using a Random Effects GLM.
- Brain Connectivity Analysis: 3 papers that perform brain-related measurements did not only analyzed brain activation, but additionally focused on the analysis of functional brain connectivity (i.e., Castelhano et al. (2018), Kosti et al. (2018), Lee et al. (2016)). For example, Castelhano et al. (2018) focuses on brain connectivity analysis and uses Granger Causality Maps. Similarly, Kosti et al. (2018) not only analyzes differences in brain activation between code comprehension and syntax tasks, but additionally examines connectivity patterns and show a topographical representation of phase interactions.

Data analysis methods applied to analyze research questions where neurophysiological data was the independent variable included supervised learning approaches (in particular classification) and supervised learning in form of clustering. Almost all studies where neurophysiological data was used as an independent variable (18 out of 19 studies, listed in Table 8) relied on supervised learning to predict dependent constructs from independent constructs (also denoted as features in this context), while one study used unsupervised learning (more specifically clustering) to predict expertise from EEG signals (Crk and Kluthe, 2014). Of the studies using supervised learning, 5 studies used feature selection methods to automatically choose the best features for classification (e.g., Couceiro et al. (2019c), Kosti et al. (2018), Medeiros et al. (2019), Müller and Fritz (2015), Züger and Fritz (2015)). Feature selection experiments investigating different combinations of features in terms of their classification accuracy were conducted by 9 studies (Behroozi and Parnin

(2018), Fritz et al. (2014), Fucci et al. (2019), Girardi et al. (2020), Lee et al. (2017), Müller and Fritz (2016), Vrzakova et al. (2020), Züger et al. (2018)). In addition, model selection experiments comparing different classification algorithms were conducted by 5 studies to identify the best classification technique for a particular setting (Behroozi and Parnin (2018), Fucci et al. (2019), Girardi et al. (2020), Züger and Fritz (2015), Züger et al. (2018)). Furthermore, a comparison of different window sizes was performed by 3 studies (Fritz et al. (2014), Züger and Fritz (2015), Züger et al. (2018)).

4.5. What are the main findings (RQ5)?

4.5.1. Neurophysiological data used as dependent variable

This section provides a synthesis of the findings of the studies that looked into the effects of various task-specific factors, developer-specific factors, team-specific factors, task performance, and context factors on neurophysiological measures. For task-specific factors, we distinguish software development activities, task-related events, and task characteristics. For an overview of the main findings, we group studies focusing on the same independent variable and discuss all studies related to an independent variable in a paragraph each. Whenever an independent variable was investigated for different software development activities, we group studies referring to the same activity together, since findings for one software engineering activity (e.g., code comprehension) cannot necessarily be transferred to another software engineering activity (e.g., code inspection).

Task-specific factors: Software Development Activities. The research reviewed in this paper provided novel insights into brain activation patterns that emerge as a result of perception of different software development activities.

The first study investigating brain activation patterns in a software engineering context was the study by Siegmund et al. (2014). For program comprehension, based on fMRI Siegmund et al. (2014) found that five different brain regions associated with working memory (BA 6, BA 40), attention (BA 6), and language processing (BA 21, BA 44, and BA 47) (all in the left hemisphere) are activated (using syntax tasks as a contrast). These findings could be largely replicated by Siegmund et al. (2017); they found activation in BAs 21, 40, and 44 within the left hemisphere; no activation of 6 and 47. Additionally, Siegmund et al. (2017) found activation in BA 39 which is related to the integration of multi-sensory information. These findings are in line with our understanding of bottom-up program comprehension. Bottom-up program comprehension tasks require participants to analyze words and symbols and integrate them to semantic chunks (using the language network BA 21, BA 44, and BA 47 and presumably BA 39 for integration) and manipulate numbers and words according to the intention of the source code, which requires keeping values of manipulated numbers and words in mind (BA 6 and BA 40) (Peitek et al., 2018a). Siegmund et al. (2017) further demonstrated that during semantic-cue comprehension and bottom-up comprehension the same brain regions (with the exception of BA 39) were activated. BA 39 is deactivated during semantic-cues comprehension, but activated during bottom-up comprehension. For all areas, the activation is significantly lower for semantic cue comprehension than for bottom-up comprehension, which confirms that beacons ease comprehension. The study additionally found that neither beacons nor program layout seem to significantly affect the program comprehension process.

Based on fMRI, Ivanova et al. (2020) aimed to disentangle brain activation due to program comprehension and brain activation that results from the underlying problem content (an aspect that has not been differentiated in the studies by Siegmund et al. (2017, 2014) and Peitek et al. (2018a)). Their results show strong bilateral responses to code comprehension tasks in the multiple demand (MD) system (i.e., regions in frontal and parietal lobes, as well as a region in the anterior cingulate cortex). These responses were significantly stronger than for sentence problems. The fact that the MD system responds more strongly to code comprehension tasks than to tasks involving textual descriptions of the code demonstrates that the response to the MD system is specific to code comprehension and not just activated due to the underlying problem content. The activation in the MD system occurs irrespective of the problem type and problem structure, can be observed across most MD regions, and generalizes across programming languages. Results further suggest that the involvement of the language system (left-lateralized activation) was mostly driven by the processing of problem content rather than code comprehension, suggesting that the language system does not support code comprehension in proficient programmers. This work further showed that code comprehension is broadly supported by the MD system. At the same time the paper found that no MD regions are functionally specialized to process source code, which is in agreement with the findings of Liu et al. (2020).

Based on fMRI, Liu et al. (2020) showed that during code comprehension (in contrast to tasks based on fake code) the lateral prefrontal cortex (middle/inferior frontal gyri, inferior frontal sulcus; mainly BA 44/46, with partial activation in BAs 6, 8, 9, 10, 47), the parietal cortex (the intraparietal sulcus, angular and supramarginal gyri; BA 7) and the posterior middle temporal gyrus and superior temporal sulcus (BA 22/37) were activated in expert programmers. Activity was also observed in early visual cortices. Fronto-parietal responses were also observed in the study by Siegmund et al. (2017, 2014) during code comprehension (in contrast to syntax tasks), by Ivanova et al. (2020) during code comprehension (when compared to textual descriptions of the code), by Huang et al. (2019) during data structure tasks, by Floyd et al. (2017) during code inspection tasks (when compared to prose review). The study further found that brain activation patterns within this network differ for different control structures (i.e., for and if statements). This work further demonstrates that in terms of underlying neural basis, code comprehension largely overlaps with other culturally derived symbol systems, in particular formal logic and to a smaller degree math. Moreover, consistent with Floyd et al. (2017) and Ivanova et al. (2020) this work showed that the neural basis of code comprehension and language is distinct. However, laterality of code and language covaried across participants, an observation that was not made by Ivanova et al. (2020).

Based on fMRI and fNIRS, Huang et al. (2019) investigated differences between mental rotation and data structure operations (i.e., tree and sequence). fMRI results show that "a number of Default Mode Network (DMN) regions involved in mental simulation were recruited more heavily during mental rotation than during code tasks; Still, 95% of voxels were statistically indistinguishable between Mental and Tree tasks" (p. 402). DMN denotes a network of brain regions that are active "when an individual is awake and alert and yet not actively engaged in an attention-demanding task" (p. 682); this network comprises, among other areas, the medial prefrontal cortex (MPFC), as well as the posterior cingulate and precuneus (Raichle et al., 2001). fNIRS results demonstrate that mental rotation and data structure operations involve activation of the same brain regions (i.e., BAs 6–9, 17–19, 39 and 46). When comparing brain activation results of sequence, tree and mental operations, only the comparison "Sequence > Mental" showed differences. This suggests that spatial ability operations and data structure operations are related. While both fMRI and fNIRS found brain activation in similar areas during both mental rotation and data structure tasks, it has to

be noted that several of the task differences observed with fMRI could not be observed with fNIRS (presumably due to the lower spatial resolution of fNIRS). The paper also reports differences in task performance between the two different methods (i.e., significantly lower task performance in terms of accuracy with fMRI). A comparison between fMRI and fNIRS with self-reports showed that the similarities between mental rotations and data structure operations (including the overlap in brain regions) were not well reflected in self-reports. What follows is that the use of brain imaging is critical to identify differences between tasks that would not have been observable with introspective methods such as surveys.

While the studies by Siegmund et al. (2017, 2014), Peitek et al. (2018a), Ivanova et al. (2020), Liu et al. (2020) and Huang et al. (2019) focussed on comprehension tasks, the studies of Duraes et al. (2016) and Castelhano et al. (2018) tried to understand brain activation patterns and connectivity patterns during code inspection. For code inspection, based on fMRI Duraes et al. (2016) found activation in several areas related to program comprehension including and beyond known language regions and areas related to working memory and decision making. At the moment of the bug detection, stronger activation was found (in the medial frontal region), while regions related to visual perception and decision-making became deactivated (right insula and bilateral occipital areas). This result suggests that the judgment about the presence of a bug had already been reached before the "bug detection" event. Moreover, Castelhano et al. (2018) showed through a connectivity analysis that evolutionary older brain regions initially used for different tasks in the history of mankind seem to be reutilized for recent and complex tasks such as code inspection.

While all of the above studies used fMRI (and additionally in one case fNIRS) two studies looked into the difference of brain activation for different software development activities using EEG. Kosti et al. (2018) showed that code comprehension (requiring mental simulation) is a more demanding task when compared to syntax tasks. In particular, increased brain activation over frontal areas (mainly the upper Beta band, [20–30 Hz]) could be observed. The differences are even more pronounced when looking at connectivity patterns. In particular, interhemispheric interactions within Theta and Upper Beta frequency bands distinguish code comprehension from syntax tasks. Moreover, the study by Kosti et al. (2018) could show that a programmer's workload correlates with brain activation patterns and the pattern of functional connectivity. More specifically, a strong correlation between brain activation in the higher bands (Lower Beta [10–20 Hz], Upper Beta [20–30 Hz], and Gamma [30–100 Hz]) and a programmer's workload could be shown. In addition, the results indicate positive correlations between couplings (i.e., phase couplings between signals) and workload within the lowest bands and within Upper Beta and significant negative correlations between couplings and workload within the lowest 4 bands. For programming tasks and bug fixing tasks, based on EEG González et al. (2015) reported that significant differences in terms of EEG compound signals both in the physical and the digital setting exist.

Task-specific factors: Task-relevant events. Duraes et al. (2016) and Castelhano et al. (2018) are the only fMRI studies in our SLR that went beyond contrasting software development activities and additionally contrasted task-relevant events. Their studies showed that initially finding a bug ("bug suspicion") and confirming a bug ("bug confirmation") led to differences in brain activation, in particular in the right anterior insula. Moreover, they showed that the activity in this region during bug intuition is positively correlated with the precision of bug detection, suggesting "that this brain region signals the quality of programmers' intuitive capacity to identify bugs when facing the inspection or analysis of challenging code". (Castelhano et al., 2018, p. 634).

Task-specific factors: Task characteristics. Numerous studies investigated the effect of task characteristics on neurophysiological measures. An overview of the results is summarized in Table 9.

It is worthwhile mentioning that most of the papers focusing on task characteristics analyzed the data at the task-level. In turn, Fakhoury et al. (2020, 2018) and Couceiro et al. (2019a) showed the potential for analyzing neurophysiological data at a more fine-grained level. More specifically, Fakhoury et al. (2020, 2018) showed that developers' cognitive load can be accurately associated with identifiers in source code and text (i.e., similarity of 78% compared to self-reported high cognitive load). Similarly, Couceiro et al. (2019a) showed that eye tracking has enough resolution to pinpoint specific code lines that correspond to moments when measurements of heart rate variability and pupillography show high levels of cognitive load.

Developer-specific factors: Expertise. For program comprehension, using EEG Lee et al. (2016) investigated differences in the information processing between novices and experts. The study showed that program comprehension involves Beta and Gamma frequencies (suggesting that high levels of concentration are used during comprehension). The results showed that the expert group experienced significantly higher power in the Beta range, presumably utilizing more brain resources. In addition, both novices and experts experienced Gamma frequency activation during program comprehension. Similarly, for program comprehension, applying EEG Crk et al. (2016) investigated the usage of EEG indicators of working memory for evaluating expertise-related differences in subject performance. The results showed significant differences in Upper Alpha (ranging from Individual Alpha Frequency (IAF)[11] to IAF+2) and Lower-1 Alpha (ranging from IAF-4 to IAF-2) for both expertise level and correctness and significant interaction effects between correctness and expertise level. For Lower-2 Alpha (ranging from IAF-2 to IAF) and Theta (ranging from IAF-6 to IAF-4) significant differences were only found for expertise level, but not correctness. Overall, the paper shows that more direct measures of cognitive load can be used to quantify comprehension task performance across different levels of expertise and that cognitive demands depend on the expertise level.

Team-specific factors: Pair Programming. Team-specific factors were in the focus of Ahonen et al. (2018, 2016). For programming, using heart- and skin-related measurements Ahonen et al. (2018, 2016) investigated the synchrony in the physiological signals during pair programming. Results show evidence for social psychophysiological compliance (SPC) in both the heart rate variability signal and the EDA signal of collaborating dyads. Moreover, self-reported task difficulty ratings were associated with SPC. In turn, heart-related measurements of physical activity (HR) were-in line with the expectations-insensitive to pair collaboration and were inconsistent with self-reports. Moreover, Ahonen et al. (2018) investigated role differences (driver versus navigator) concerning task-relevant events (i.e., running and testing code). The results show that failure events led to a significantly higher skin conductance response (SCR) for developers in the driver role, which "might reflect engagement, or the liability of being in the leading role" (Ahonen et al., 2018, p. 8). Moreover, they found that for drivers the SCR increases up to event time, while the physiological response of the navigator is delayed by several

[11] A fixed Alpha frequency band (8–14 Hz) might not cover all Alpha activity, since peak Alpha frequency differs between individuals. Individual Alpha Frequency (IAF) can be calculated for each individual and frequency bands can then be defined based on IAF dynamically.

Table 9
Summary of findings for studies on task characteristics.

Task difficulty		
fNIRS (Oxy-Hb)	Normalized Oxy-Hb significantly larger for hard vs. easy tasks	Nakagawa et al. (2014)
fNIRS	A comparison of hard and easy tasks did not yield any significant effects	Ikutani and Uwano (2014)
fMRI and fNIRS	- increasing brain activation with increasing task difficulty using fMRI; significantly larger effect for difficult sequence tasks than for mental rotation tasks - no significant effects for fNIRS	Huang et al. (2019)
EEG	Task difficulty is reflected in the EEG signal	Duraisingam et al. (2017)
EEG	Significant higher Alpha and Theta magnitudes for confusing code snippets compared to non-confusing ones	Yeh et al. (2017)
Eyetracking (pupillography)	Programmers' cognitive load measured using pupillography is consistent with the subjective perception of task difficulty	Couceiro et al. (2019b)
Eyetracking (pupillography), Heart	Eye tracking has enough resolution to pinpoint specific code lines that correspond to moments when measurements of heart rate variability and pupillography show high levels of cognitive load.	Couceiro et al. (2019a)
Structural inconsistencies and linguistic antipatterns		
Eyetracking and fNIRS	- developer's cognitive load can be accurately associated with identifiers in source code and text, with a similarity of 78% compared to self-reported high cognitive load - linguistic antipatterns in the source code led to a significant increase of the cognitive load experienced by the participants - no statistical effect of structural inconsistencies on average cognitive load (however participants report frustration) - source code containing both lexical and structural inconsistencies misled 60% of the participants - participants who successfully completed the task experienced higher cognitive load when both inconsistencies were present	Fakhoury et al. (2020, 2018)
Problem Structure		
fNIRS	Contrasting numeric, variable, and control tasks shows significantly higher brain activation in condition "Variable" compared to the other two conditions for problem type.	Ikutani and Uwano (2014)
Semantic-cues comprehension and bottom-up comprehension		
fMRI	- for all areas involved during program comprehension, the activation is significantly lower for semantic cue comprehension than for bottom-up comprehension, which confirms that beacons ease comprehension - neither beacons nor program layout seem to significantly affect the program comprehension process	Siegmund et al. (2017)
fMRI	In line with our understanding of bottom-up comprehension data-flow complexity and vocabulary size are highly correlated with the concentration level needed for program comprehension (i.e., there are no beacons that could act as cues for data-flow aspects).	Peitek et al. (2018a)
Interview Setting		
Eyetracking (eye blinks)	- whiteboard setting perceived as more stressful by participants than the paper setting - significant differences between settings for several measures associated with stress and cognitive load (incl. longer blinks)	Behroozi et al. (2018)
Code regularity		
Eyetracking (pupil size)	Descriptive statistics show a decreasing tendency of visual effort (for fixation count and total fixation duration, but less clearly for pupil dilation) for repeated instances	Jbara and Feitelson (2015)
Presence/absence of structural and textual features		
Eyetracking (pupil size)	No significant differences between conditions could be observed in terms of pupil size. The analysis of questionnaire data showed that the condition combining both structural and textual elements was significantly more comprehensible and readable than the three other conditions.	Wulff-Jensen et al. (2019)
Attention data representation		
(pupil size)	No significant effect of attention data representation on efficiency and quality of software maintenance tasks (including cognitive load).	Ahrens et al. (2019)
Programming languages		
EEG	Initial exploration of differences in brain activation dependent on the programming language using descriptive statistics	Doukakis (2019), Doukakis et al. (2020)

seconds. For pass events the SCR decreases for the driver before the event and it then rebounds once the driver relaxes. In turn, for the navigator, an increase after the pass event occurs. In addition, using EEG Ikramov et al. (2019) conducted an initial exploration of the influence of the programming role (i.e., solo, pair/navigator, and pair/driver) on brain activation. For the condition, Navigator, a significantly higher activity in the L1 Alpha sub-band could be observed. No significant differences in the other Alpha sub-bands and the Theta band could be observed.

Task Performance. For programming, Yamamoto et al. (2016) showed the usage of the EEG signal as an index for classification of programmers who fail to find an implementation strategy. The results show that the EEG signal *during* task execution contains a significantly larger Alpha wave power and Beta/Alpha ratio when successfully finding an implementation strategy. The EEG signal *after* task execution contains a significantly larger Alpha wave power when successfully finding an implementation strategy. When analyzing the effect of individual differences, the analysis showed that more than half of the participants had a significantly larger Alpha wave and Beta/Alpha ratio when they succeeded in finding an implementation strategy. In a related study, using EEG, Ishida and Uwano (2019a) investigated differences in brain activation depending on the success in performing a code comprehension task and obtained similar findings. Results show that both Alpha and Beta power significantly increased between start and end of the code comprehension task for developers that could successfully complete the task. For the failure group, the difference in brain activation between start and end was not significant. In addition, Ishida and Uwano (2019b,a) showed an earlier increase of Alpha activity in the success group. Similarly, for code inspection Ishida and Uwano (2019a) showed that both Alpha and Beta power for the success group significantly increase. For the failure group a significant increase of Beta, but not Alpha power could be observed.

Context Factors: Listening to Music. Using EEG, Ikramov et al. (2019) explored the effect of programming with and without music in an initial study. Results remained inconclusive. Readers interested into the effects of music in the context of software development and information systems are referred to a recent paper by Gefen and Riedl (2018).

4.5.2. Neurophysiological data used as independent variable

This section provides a synthesis of the findings of studies that looked into the efficacy of neurophysiological measures to predict different dependent variables (e.g., task-specific factors, developer-specific factors, mental states, and task performance). Appendix F provides an overview of the measures the different studies used as features for their machine learning models. A comparison of the different classifiers in terms of their performance is presented in Appendix H. For a better comparison of the findings, we group studies focusing on the same dependent variable and discuss all studies related to one dependent variable in one paragraph. When a dependent variable was investigated for different software development activities, we group studies referring to the same activity together. The rationale for this approach is that findings for one software engineering activity (e.g., code comprehension) cannot necessarily be transferred to another software engineering activity (e.g., code inspection).

Task-specific factors: Software Development Activities. Based on fMRI, Floyd et al. (2017) showed that brain activation patterns of programming languages and natural languages are distinct; code review, code comprehension and prose review have largely distinct activation patterns (a finding which is in line with Ivanova et al. (2020)). They further demonstrated that the classification of these tasks based on brain activity alone is possible. Moreover, the study showed that a number of prefrontal regions reliably distinguished between code and prose. For experienced programmers, code and prose were hardly distinguishable, which signifies that programming languages are increasingly treated like natural languages with increasing expertise. Similarly, based on EEG, skin- and heart-related measurements Fucci et al. (2019) showed that code and prose comprehension can be accurately differentiated. The best performing classifier reached an accuracy of 87% and was based on heart-related measurements. However, in contrast to the study by Floyd et al. (2017) no relationship between classification accuracy and expertise was found.

Task-specific factors: Task Category and Task Subcategory. Based on fMRI, Ikutani et al. (2020) showed that brain activation patterns for functional categories of source code can be distinguished during program categorization tasks. Results further showed that classification accuracies for various regions in the frontal, parietal, and temporal cortices were significantly linked to task performance. In addition, classification accuracies of subcategories on the left supramarginal gyrus and superior temporal gyrus regions were associated with task performance. Since task performance was also highly correlated with expertise, this suggests that cortical representations of functional categories (subcategories) might be associated with advanced-level programming expertise.

Task-specific factors: Problem Structure. While Ikutani et al.'s (2020) fMRI study showed that task categories and subcategories can be distinguished, Liu et al. (2017) took a more fine-grained perspective and focused on different structures within a program. Based on fMRI, Liu et al. (2020) showed during program comprehension that patterns of activity within the lateral prefrontal cortex (accuracy 64.7%), intraparietal sulcus (accuracy 67.4%), posterior middle temporal gyrus (accuracy 68.4%), and to a lesser extent within the early visual cortex (accuracy 55.7%) distinguished between for and if statements.

Developer-specific factors: Expertise. For code comprehension, expertise is predicted by Lee et al. (2017) using EEG and eye tracking. The best overall performance could be achieved by using EEG and eye tracking features in combination using a SVM classifier (97.7% precision and 96.4% recall). Similarly, Crk and Kluthe (2014) used EEG (more specifically Upper Alpha and Theta) to predict the expertise of a developer during code comprehension tasks. The accuracy for detecting expertise ranged between 55% and 59% (and 56% and 67% when considering only correct answers). Therefore, compared to Lee et al. (2017), classifier performance is considerably lower. While both studies focused on a 2-state classification of expertise, the participants in the study by Lee et al. (2017) had clearly different levels of expertise (novices with 1 year of programming experience versus experts with at least 6 years of experience). In contrast, Crk and Kluthe (2014) examined students that differed in their class level, most likely signifying a much lower level of expertise variance. Differences in the used measurement instruments (i.e., Lee et al. (2017) used a research grade EEG device, while Crk and Kluthe (2014) used a consumer-grade device) and differences in the used neurophysiological methods (i.e., Lee et al. (2017) used multimodal measurements, while Crk and Kluthe (2014) used a single modality) probably also contributed to the observed differences in classifier performance.

Mental State: Cognitive Load. Several studies developed classifiers for cognitive load using different neurophysiological methods. More specifically, EEG, heart, skin, and breath-related measurements as well as measurements of pupil size were used along with behavioral measures like eye gazes, code metrics, interaction metrics, and change metrics.

Classifiers for predicting cognitive load (2-state classification) during code comprehension tasks based on a single modality are proposed by Duraisingam et al. (2017), Couceiro et al. (2019c), and Medeiros et al. (2019). Couceiro et al. (2019c) predicted cognitive load (2-state classification) during code comprehension using heart-related measurements (more specifically HRV), while Medeiros et al. (2019) used EEG. In both studies, code snippets differing in terms of code complexity were used as stimuli. Results show that neurophysiological measurements of cognitive load correspond to subjective load perceptions of the participants, but not with the code complexity of the stimuli. These findings are in line with the fMRI work of Peitek et al. (2018a) who examined the association of different complexity metrics and brain activation. While several metrics were tested only in one case, could a significant correlation be established (i.e., DepDegree). In the light of cognitive load theory (Sweller, 2011) which differentiates between intrinsic load (inherent complexity of the task) and extraneous load (complexity stemming from the task representation), such discrepancies in study findings are not surprising. Existing metrics typically only capture some of the aspects that are known to contribute to the cognitive load perceived by a developer. Additional research is needed to better understand which metrics, or combinations of metrics, are best suited to capture what is cognitively demanding and what is not.

Similar to Medeiros et al. (2019), the classifier proposed by Duraisingam et al. (2017) is based on EEG. A comparison of different feature groups shows that the best results were obtained by considering interhemispheric differences (i.e., asymmetry ratio). In turn, Lee et al. (2017) uses a combination of EEG and eye tracking. The best overall performance (for a 2-state classification) was achieved by using EEG and eye tracking features in combination. A classifier for predicting cognitive load (2-state classification) during code comprehension based on low-cost, off-the-shelf sensors (EEG, EDA, eye tracking) was developed by Fritz et al. (2014). The authors tested the performance of the classifier when predicting a new participant, a new task, and a new participant–task pair. Their results showed that different combinations of sensors performed best depending on the setting. The study found that precision and recall for predicting new participants were 15% and 5% lower than for predicting new tasks. Interestingly, in contrast to several other studies (e.g., Lee et al. (2017) and Müller and Fritz (2016)), the combination of all sensors did not always lead to the best classifier performance. Thus, it is *not* possible to establish the general rule that a combination of different physiological indicators is always a better predictor of a dependent variable than one physiological measure alone.

While all of the above studies focused on a 2-state classification, Kosti et al. (2018) developed a 4-state classifier for cognitive load using a consumer-grade EEG. The results show that functional connectivity measures better express the relation between the difficulty of a comprehension task and the workload of a programmer, if compared to measures of signal power in the different frequency bands. Moreover, the functional connectivity approach offered an additional advantage: it assessed cognitive load independent of the participant, a property that increases practical applicability.

While all of the previously mentioned studies focused on code comprehension, Müller and Fritz (2016) focused on general software development activities and developed classifiers for predicting cognitive load (6-state classification) from neurophysiological data (heart, skin, and breath) as well as code, change and interaction metrics at both the class and the method level. The evaluation showed that the best results could be achieved by combining all metrics. The classifier using neurophysiological data outperformed the classifier using interaction, code, and change metrics in 3 out of 4 cases. In a replication, confirming the original study results, the classifier using all features performed best. The comparison of neurophysiological data with interaction, code, and change metrics again yielded mixed results. The findings further show that the classifier is highly sensitive to the individual subject. Specifically, it was found that when training the classifier based on the data of all subjects, predictions were not better than chance.

In general, the overlap of features between studies is rather limited. Thus, additional research is needed to advance our understanding of which feature combination works best in different software engineering contexts.

Mental State: Affective State. A classifier that can distinguish positive and negative emotions (i.e., valence) based on several features combining EEG, eye tracking, skin and heart-related measurements during change tasks was developed by Müller and Fritz (2015). Similarly, Girardi et al. (2020) developed a classifier for valence using EEG as well as skin- and heart-related measurements during a change task. The performance of the classifier only using skin and heart-related measurements as provided by the Empatica E4 wristband was comparable to a classifier additionally using EEG. While the results in terms of classifier performance are comparable with Müller and Fritz (2015), they are achieved using a smaller sensor set. In addition to valence, Girardi et al. (2020) additionally developed a classifier for arousal. Again, the performance of the classifier only using the Empatica E4 wristband (i.e., using skin- and heart-related measurements) was comparable to a classifier additionally using EEG.

The development of a classifier for both valance and arousal during code inspection combining skin-related measurements with behavioral data (touch and eye gazes) was in the focus of Vrzakova et al. (2020). For both valence and arousal the best performance could be achieved by a classifier combining all modalities (valence: accuracy = 90.0%, arousal: accuracy = 83.9%). The results also show that positive valance could be better predicted when compared to negative valence (as indicated by a true positive rate of 95.7% when compared to a true negative rate of 69.5%). The best classifier using one modality alone was based on eye gaze (valence: accuracy = 85.8%, arousal: accuracy = 76.6%). Classifier performance is superior when compared to the work by Müller and Fritz (2015) and Girardi et al. (2020), however, it needs to be kept in mind that Müller and Fritz (2015) and Girardi et al. (2020) focused on change tasks, while Vrzakova et al. (2020) focused on code inspections. In addition, Vrzakova et al. (2020) showed that affect builds up over time. More specifically, it was found that when using data from the beginning of the task, compared to data from the end of the task, performance of the classifier combining all modalities decreased by 4% for both valence and arousal.

Mental State: Interruptibility. Interruptibility could be classified with high accuracy into two states for general software development activities (both in a lab and a field study) using EEG, skin and heart-related measurements (Züger and Fritz, 2015). Features that proved useful for a 2-state and 5-state classification in both studies include activity in the EEG's beta and gamma frequency band as well as mean skin temperature. In

turn, Züger et al. (2018) focused on predicting interruptibility based on heart-related measurements as well as behavioral data (computer interactions, sleep, and physical activity) in the field. Results indicate that interaction metrics slightly outperformed heart-related measures. The best results were achieved by a classifier combining interaction metrics and heart-related measures. The study also showed that the optimal time window varies per feature. Furthermore, the paper demonstrated that even a generally trained model can accurately predict interruptibility for new subjects. This is an important result, because the possibility to use a machine learning model which was not specifically trained for a particular subject significantly increases practical applicability. The results of our review show that for other mental states, like cognitive load, classifiers were often sensitive to the individual. For example, Müller and Fritz (2016) report that their classifier trained on the data of all participants did not work better than chance.

Mental State: Stress. Stress was classified using eye tracking measures (including both neuro-physiological and behavioral measures) during general software development activities (Behroozi and Parnin, 2018). The best performing classifier, irrespective of the applied labeling strategy, was Random Forest. Classifier performance was best when considering the interview setting (i.e., paper or whiteboard) along with the stress rating in the labeling (accuracy 88%). This implies that eye tracking measurements show stress differently as a function of the setting. Therefore, the study results are hardly surprising as most of the measures that were considered as features in the machine-learning model were based on eye movements (which were likely influenced by the setting). The results further showed that fixation-based measures, as well as pupil size, are more predictive than saccadic measurements.

Mental State: Perceived Progress. A classifier that can distinguish high from low progress was developed by Müller and Fritz (2015) based on multi-modal measurements combining features from EEG, eye tracking, and EDA. High and low progress could be distinguished in 67.70% of all cases. Change in Alpha activity, change in Beta/Theta ratio, change in mean temperature peak amplitude, maximum pupil size, change in mean pupil size, and change in mean skin conductance level were the most predictive features. Change in Alpha activity and change in Beta/Theta ratio were also shown to be among the most predictive features for distinguishing positive and negative affect.

Task performance. While Müller and Fritz (2015) focused on perceived progress, Crk and Kluthe (2016) predicted task performance during code comprehension tasks. The study found that both the individual alpha frequency (IAF)[12] and programming experience play a statistically significant role. The results further show that experience increased the likelihood of correct answers substantially more than IAF.

Presence of Quality Concerns. For general software development activities, Müller and Fritz (2016) developed a classifier for predicting quality concerns from neurophysiological data (heart, skin, and breath) and compared this physio-classifier with classifiers using interaction, code, and change metrics. The results show that for a within subject setting the classifier based on neurophysiological data outperformed the classifier based on code and interaction metrics as well as the classifier combining all features. For the across subject setting, in turn, the classifier based on all data sources performed best.

[12] A fixed Alpha frequency band (8–14 Hz) might not cover all Alpha activity, since peak Alpha frequency differs between individuals. Individual Alpha Frequency (IAF) can be calculated for each individual.

5. Discussion and impact

5.1. Who published NeuroSE research and where (RQ1)?

Based on the analysis of N = 89 NeuroSE publications, we identified 191 different authors. A recent review of the NeuroIS literature found that 432 different authors published 200 NeuroIS articles (Riedl et al., 2020a). Using this finding from NeuroSE's major sister discipline as benchmark, we can conclude that the group of researchers involved in NeuroSE research is still relatively small. Moreover, our results show that there is a small group of authors that is highly active (14 authors are involved in more than 28% of the NeuroSE publications). It follows that there is a notable inequality in research contributions. The small absolute number of highly engaged researchers together with the observed inequality can be seen critical. One major consequence of this still low number of highly engaged NeuroSE researchers is that potential candidates to serve as editors and reviewers of NeuroSE papers are a scarce resource. However, our list of top-14 contributors (cf. Fig. 2) can be used as a basis to select associate editors and reviewers for corresponding publications.

Our analysis of outlets shows that more than 40% of the NeuroSE publications appeared in five outlets (EMIP, ICSE, ICPC, SEmotion and FSE) and over 60% of the NeuroSE publications concentrated on the 13 outlets (cf. Table 3). The remaining papers are distributed across more than 30 further outlets. So far, only three journals published more than one NeuroSE paper (i.e., Information and Software Technology, Empirical Software Engineering, and eLife). The low ratio of journal papers is an indication that the field is still rather nascent. At the same time, since 2014 five NeuroSE papers were published in ICSE, which underlines the relevance of the topic and highlights that several high quality contributions already exist. The increasing attention on neurophysiological measurements at workshops like EMIP or SEmotion, or at conferences like ICPC, makes us confident that we will see more conference and journal papers with a NeuroSE focus in the near future.

5.2. What kind of NeuroSE research was published (RQ2)?

Our results indicate that – except 2019 – there was no year in which more than ten completed empirical research papers with a NeuroSE focus were published (see Fig. 3, 2019, blue bar). This shows that the field of NeuroSE is still at a relatively nascent stage. Our literature review also identified methodological contributions. This includes recent work by Peitek et al. (2018d) investigating the integrated collection of fMRI and eye tracking data, the introduction of an infrastructure to collect eye tracking data linked to the software artifact (Guarnera et al., 2018), and a novel tool for multi-modal data exploration during code comprehension experiments (Peitek et al., 2019). With the exception of recently proposed methodological guidelines for conducting eye tracking studies (Sharafi et al., 2020), we find that what is missing so far are papers providing methodological guidelines on how to conduct NeuroSE research based on specific methods. Such papers should consider the idiosyncrasies of the field. The NeuroIS field may serve as an example. A number of methodological papers have been published in this field, including EEG guidelines (Müller-Putz et al., 2015), fMRI guidelines (Dimoka et al., 2012), fNIRS guidelines (Gefen et al., 2014), or eye tracking guidelines (Djamasbi, 2014); and even more specific papers on analysis techniques such as those related to network analysis of brain imaging data have been published in this field (Hubert et al., 2017). In this context, another fruitful avenue for future work is to focus on methodological contributions related to specific themes. Again, the NeuroIS field may serve as an example. In

this field, methodological contributions related to technostress (i.e., stress caused by the use and ubiquity of digital technologies) have been published, such as a paper on blood pressure measurement (Fischer et al., 2017) and an article on heart rate variability (Baumgartner et al., 2019).

The thematic focus of the majority of studies was on code comprehension (30 of 47). This is not surprising since code comprehension forms the basis of other software development activities like, for example, code inspection or change tasks. While earlier fMRI studies focused on code comprehension and to a smaller extent code inspection (Castelhano et al., 2018; Duraes et al., 2016), and hence contributed to a better understanding of the underlying cognitive processes, more recent fMRI studies have focused on the question in how far code comprehension is distinct from other culturally derived symbol systems like math, logic, or language (Ivanova et al., 2020; Liu et al., 2020) (cf. Table 4). We expect that fMRI studies will continue to focus on code comprehension and code inspection, because participants are highly restricted in their motor movements during fMRI studies, and code comprehension and code inspection tasks (unlike programming or change tasks) can be designed in such a way that user interactions are minimized. However, this focus on code comprehension and code inspection was also apparent when other neurophysiological methods were used. In particular, methods related to ANS activity (as shown for example by Ahonen et al. (2018, 2016), Müller and Fritz (2016), Vrzakova et al. (2020), Züger and Fritz (2015), Züger et al. (2018)) have the potential for being used in real-world settings to assist programmers during various software development activities. With the ongoing maturity of the field we expect corresponding contributions and hence a more diverse focus on different software development activities can be expected in the future. Several studies compared brain activation between different software development activities (e.g., code comprehension and code inspection) and could show corresponding differences in brain activation patterns (e.g., Floyd et al. (2017)). This constitutes evidence that the mental processes underlying different SE activities vary. Moreover, this insight suggests that we cannot expect that classifiers for mental states trained on one software development activity generalize towards another software development activity. Thus, replication studies are needed to test the usage of the developed classifiers in different settings.

5.3. Which methods and measures did existing NeuroSE publications apply (RQ3)?

Our literature review revealed that brain activity (fMRI, EEG, fNIRS) was studied more frequently than ANS activity (heart rate, skin conductance, pupil dilation). Moreover, our results indicate that EEG is the dominant method in NeuroSE research (20 out of 47 papers; 43% of the completed empirical studies used EEG), followed by fMRI (10 papers; 21%), eye tracking (10 papers; 21%), heart-related measurements (10 papers; 21%), skin-related measurements (8 papers; 17%), fNIRS (4 papers; 9%), and measurement of respiration (1 paper; 2%). The mentioned review of the NeuroIS literature by Riedl et al. (2020a) also identified EEG as the most frequently applied method. One major explanation for this finding is that consumer-grade EEG measurement devices are increasingly available on the market, motivating researchers to explore the usage of this method in an SE context.

The consequence of this development is that researchers can measure brain activity at low cost and with relatively little effort (e.g., because the application of electrodes is usually much more time-consuming with research-grade instruments). However, whether or not these consumer-grade EEG devices offer the necessary reliability and validity is a topic of ongoing discussion (Riedl et al., 2020b). In our review we identified 20 completed empirical EEG studies, six of which used Emotiv EPOC (a consumer-grade instrument). Other consumer-grade tools which were used are NeuroSky mindset headset (three studies), a NeXus 10 MARK II (three studies), a NeuroSky mindwave headset (one study), and a BrainLink Pro headset (one study).

Some validation studies exist which suggest that these low-cost EEG tools may offer measurement quality similar to high-end research devices. For example, Badcock et al. (2013) showed that Emotive EPOC may prove a valid alternative to research-grade EEG tools like NeuroScan for recording reliable auditory event-related potentials. Moreover, Sánchez Reolid et al. (2018) showed that Emotiv EPOC+ can be used with high confidence to classify the emotional state of a user. Other researchers, based on empirical evidence, are less optimistic. For example, Duvinage et al. (2012) write that "the Emotiv headset performs significantly worse than the [ANT system, Advanced Neuro Technology]" (p. 1). Based on this finding, Emotiv should only be used "for non-critical applications such as games" (Duvinage et al., 2012, p. 1). This clearly highlights the need for future methodological studies that compare, in different SE contexts, low-cost EEG tools with research-grade instruments in different SE contexts to establish an enhanced understanding of the measurement properties of different instruments in different contexts. We make a call NeuroSE researchers to actively contribute to this discussion, also because this issue not only pertains to EEG as method, but also to other measurement instruments, such as heart rate measurement based on smart watches (if compared to high-end ECG devices). As a starting point for EEG, please refer to Riedl et al. (2020b).

While most studies only applied one neurophysiological method, the studies by Ahonen et al. (2018), Couceiro et al. (2019a), Fritz et al. (2014), Fucci et al. (2019), Girardi et al. (2020), Huang et al. (2019), Müller and Fritz (2015, 2016), Züger and Fritz (2015) were multi-modal and hence combined several neurophysiological methods. In addition, the studies by Castelhano et al. (2018), Couceiro et al. (2019a), Fakhoury et al. (2020, 2018), Ishida and Uwano (2019b), Lee et al. (2017), Vrzakova et al. (2020), Züger et al. (2018) combined one or several neurophysiological method(s) with behavioral data. Moreover, the studies by Ahrens et al. (2019), Aschwanden and Crosby (2006), Behroozi et al. (2018), Behroozi and Parnin (2018), Jbara and Feitelson (2015), Wulff-Jensen et al. (2019) combined pupil size and/or blink rate with behavioral eye tracking metrics (i.e., fixation-based measures and/or saccadic measures). The collection of multi-modal neurophysiological data is likely to increase in the future, since one modality often balances the limitations of another modality regarding measurement of a specific theoretical construct like cognitive load (Işbilir et al., 2019). In this context, diagnosticity ("a property of a measure that describes how precisely it captures a target construct as opposed to other constructs" (Riedl et al., 2014, p. 29)) and confounding factors are essential. For example, measurements of pupil dilation are related to attention *and* cognitive load, among other constructs, and are influenced by various context factors, such as ambient light (Duchowski et al., 2018). In turn, EEG offers high temporal resolution, but to achieve high spatial resolution, EEG instruments with a sufficient number of electrodes and algorithms which draw upon different neurophysiological assumptions are required (Müller-Putz et al., 2015). Moreover, artifacts due to eye movements, muscular movements and heart beats pose challenges to the use of EEG in the field (Işbilir et al., 2019). fNIRS provides a balance of spatial and temporal resolution. The latency of the signal, however, might pose limits to the method. Research has shown that the combination of multiple modalities may lead to

superior results when compared to single modalities (e.g., Aghajani et al. (2017), Liu et al. (2017)), yet this is not necessarily the case.

The benefits of combining different modalities are supported by several of the studies included in our review. For example, the results described in Lee et al. (2017) showed that combining EEG and eye tracking features led to more accurate predictions of task difficulty and expertise when compared to using each modality alone. Similarly, the results described by Müller and Fritz (2016) and Vrzakova et al. (2020) suggest that the combination of modalities can improve classifier performance. The complementary strength between fMRI and fNIRS, in turn, is discussed in Huang et al. (2019). While both fMRI and fNIRS found brain activation in similar areas during both mental rotation and data structure tasks, several of the task differences observed with fMRI could not be observed with fNIRS, which might be a result of the lower spatial resolution of fNIRS, and its property that areas located deep in the brain (such as limbic areas mainly related to affective information processing) cannot be studied at all. In turn, the usage of fMRI led to a significantly lower task performance in terms of accuracy (Huang et al., 2019), which presumably stems from its higher intrusiveness ("the extent to which a measurement instrument interferes with an ongoing task, thereby distorting the investigated construct [whereat the] three major dimensions of intrusiveness are degree of movement freedom, degree of natural position, and the invasiveness of an instrument" (Riedl et al., 2014, p. 29). While benefits of multimodal experiments in providing a more holistic understanding have been shown in several studies, tool support is still scarce. This need is taken up by the work of Peitek et al. (2019) proposing CodersMuse, a tool for exploring multi-modal data during program comprehension experiments. Similarly, Roy et al. (2020) proposed a tool called VITALISE for combining physiological data and eye tracking data.

The results of the literature review also show that there is a tendency to explore the combined use of neurophysiological data and eye tracking. For example, Ishida and Uwano (2019b) explored differences in brain activation and eye movements over time depending on task performance. Moreover, Castelhano et al. (2018) used eye tracking to determine fixations inside and outside of bug AOIs (Area of Interest) with the goal to control for visual attention. Moreover, Peitek et al. (2018d) explored the combined collection of fMRI and eye tracking data in order to detect fixations at the level of source code identifiers. In both cases eye tracking data (more specifically fixations) are used to enable a more fine-grained analysis of the fMRI BOLD signal. Similarly, Fakhoury et al. (2020, 2018) demonstrated that a developer's cognitive load can be accurately assessed using fNIRS and eye tracking. Couceiro et al. (2019a) explores the usage of eye tracking together with HRV to identify code lines (and even lexical tokens inside of code lines) that correspond to high cognitive load. We expect to see more studies going in this direction, since the combined usage of eye tracking with neurophysiological measurements allows researchers to link neurophysiological data with the software artifact the subject is looking at and enables a fine-grained analysis of a developer's mental state. One notable example in this domain is a paper which introduces the eye-fixation related potential (EFRP) method. This method allows one to synchronize eye tracking with EEG recording to precisely capture a person's neural activity at the exact time at which he or she starts to cognitively process a stimulus (e.g., event on the screen) (Léger et al., 2014).

5.4. How was the empirical NeuroSE research conducted (RQ4) and what are the main findings (RQ5)?

Study Participants. Our review shows that the mean sample size is 17.00, ranging from 2 to 70 participants. In comparison, Riedl et al. (2020a) found an average sample size of 45 subjects (min: 5, max: 166, median: 30, SD: 35) in a review of NeuroIS papers. Thus, the average sample size in NeuroIS research is significantly larger than in NeuroSE research. However, sample sizes of the fMRI studies in our review (mean: 18.00, 10 studies) are almost identical to the NeuroIS review (17.9, 11 studies) (Riedl et al., 2020a). Moreover, according to Riedl et al. (2017b) these numbers are comparable with sample sizes in brain imaging studies including investigations published in prestigious journals such as Neuron, Science, and Nature (average $N = 18$). For all other methods the sample sizes we observed are considerably below the ones reported for the field of NeuroIS (Riedl et al., 2020a). In particular, mean sample sizes of EEG studies (min: 2; mean: 10; max: 38) are very small, indicating overall relatively low maturity in this specific methodological domain. The studies included in the review range from studies that mostly explored the potential of using EEG in a SE context to studies conducting advanced analyses including connectivity measures. What follows is that some of the EEG studies constitute high-quality work. Our review also showed that study participants were predominantly male with a mean of 13.39% female participants. This distribution is far from balanced. However, it has to be noted that this roughly corresponds to the gender distribution in typical Computer Science programs (Huyer, 2015). Because experimental research often draws upon student samples, our finding regarding gender distribution in the sample is not surprising. Importantly, as NeuroSE research moves from the laboratory to the field, it is hoped that the rate of female subjects will become larger.

Stimuli. Our analysis showed that the stimuli used (i.e., code snippets) were relatively small and ranged for most studies from just a few lines up to 60 lines. One exception is the study by Ahrens et al. (2019), which is considerably larger with over 900 lines. In particular, when using measurement methods like fMRI where participants are highly restricted in their movements, the usage of small code snippets that can be read without substantial need for navigation is reasonable. At the same time, however, tasks involving just a few lines of code do not represent the complexity a professional developer encounters when developing software. Combined with the observation that except for six studies all were conducted in the lab, our results suggest that their emphasis was on controlled settings and that their focus was less on ecological validity than on internal validity. The usage of methods related to ANS activity (e.g., heart-related measurements, skin-related measurements, eye tracking), along with mobile EEG, has the potential to complement the existing lab studies with findings from real-world settings (i.e., Ahonen et al. (2018, 2016)), Müller and Fritz (2016), Vrzakova et al. (2020), Züger and Fritz (2018)).

To fully exploit the potential of neurophysiological measurements, the combined use with behavioral data seems highly promising. Recent research on eye tracking in the context of SE increasingly exploits the benefits of linking eye-tracking data to the software artifact. For example, Abid et al. (2019) present an eye tracking study on reading and summarizing Java methods. Unlike previous studies that only used short Java methods in isolation, this study had access to all source code (using iTrace Guarnera et al., 2018) and subjects could freely scroll and navigate between files. Following a similar approach, future research could link neurophysiological measures with the software artifacts, or other behavioral datasets such as clickstream or mouse data. For example, using a combination of user interactions and

eye tracking data Burattin et al. (2019) were able to predict (during the creation of a conceptual model) the task in which a user was engaged at a specific point in time. Information about what the user is currently working on allows contextualization of the observed mental states and can give raise to the development of neuro-adaptive systems that support the user in a context-specific manner. As an example, Adam et al. (2017) presented a blueprint for stress-sensitive adaptive enterprise systems. A major characteristic of such systems is that neuro-signals (e.g., heart rate or skin conductance) are integrated as real-time stress measures, with the goal that systems automatically adapt to the users' stress levels, thereby improving human–computer interactions. A major source for corresponding research is *IEEE Transactions on Affective Computing*. We refer the reader to this outlet to learn more about the foundations and applications in the domain of neuro-adaptive systems.

Experimental design. Most of the studies used a within-subject design with repeated measurements. This is not uncommon for neurophysiological experiments, since the range of differences across individuals is, for most neurophysiological measures larger than the range of expected changes as a result of a stimulus (Jennings and Allen, 2016). This can also be a challenge for the development of robust classifiers of mental states (e.g., frustration, cognitive load) that ideally should work with high accuracy not only for the subjects that were used for training, but also for new subjects.

Dependent and independent constructs. The results of the review show that task-specific factors (e.g., software development activity, task-related events, task characteristics) were primarily used as independent variables when neurophysiological data was used as a dependent variable (cf. Fig. 6). An example question, yet one of the most fundamental ones, is "What happens in the brain, or the nervous system, while a developer is engaged in a specific SE task?". Importantly, such brain processes are dependent on an individual's expertise. The study described in Floyd et al. (2017) is one of the few studies considering developer-specific factors like expertise (collected via subjective measurements). The paper showed that for experienced programmers Code and Prose were hardly distinguishable. It follows that source code is increasingly treated like a natural language with increasing expertise. This suggests that papers looking into brain activity should consider expertise in their research models (in particular when study subjects differ in their level of expertise). Similarly, when developing classifiers for mental states like perceived task difficulty, or cognitive load, the inclusion of developer-specific factors like expertise seems critical, since individuals differ in how efficiently they can make use of their working memory capacity and other cognitive resources.

While several studies focused on cognitive processes and the prediction of cognitive states like cognitive load or interruptibility, so far only three studies (i.e., Girardi et al. (2020), Müller and Fritz (2016), Vrzakova et al. (2020)) present completed research using neurophysiological measurements with a focus on emotional processes. Yet, emotions in an SE context have received increasing attention in recent years. For example, emotions in SE are in the focus of the International Workshop on Emotion Awareness in Software Engineering (SEmotion). Six of the papers published at SEmotion either discussed the usage of neurophysiological measurements at a conceptual level or presented suggestions for study designs. Moreover, sentiments and emotions in SE were the focus of a recent IEEE Software special issue (Novielli and Serebrenik, 2019). Similarly, a recent Journal of Systems and Software special issue had affect awareness as its focus (Novielli et al., 2018). The papers of these special issues primarily relied on natural language processing and sentiment analysis for detecting affective states. In addition, a recently published review by Sánchez-Gordón and Colomo-Palacios (2019) focuses on emotions in SE. The review showed that neurophysiological data for measuring emotions has only obtained limited attention so far. This increasing interest in the role of emotions in SE is an opportunity for the field of NeuroSE, particularly when it comes to the detection of affective states. Future research is needed that investigates the detection of emotions using neurophysiological data in comparison and combination with other modalities. The paper by Vrzakova et al. (2020) is a first example in which neurophysiological and behavioral measurements for affect prediction are compared (i.e., skin-related measurements, eye gazes, and touch). The results showed that the classifier based on eye gaze data was able to achieve the best performance. The potential of neurophysiological methods for recognizing and inducing emotions during programming has also been investigated in recent work by Girardi et al. (2020). A section on "emotion research" in a recent research agenda paper may serve as a starting point for future research. In particular, we stress that conceptual clarity about the meaning of emotion, and how it differs from related concepts such as feelings and affect, is critical to move the discipline forward with the necessary scientific rigor. This issue is discussed in a research agenda paper that pertains to the NeuroIS field (vom Brocke et al., 2020), but the arguments presented in this paper are directly relevant to the NeuroSE field too.

Data analysis and Findings. Our data analysis showed that most of the studies related to brain activity (fMRI, EEG, fNIRS) focused on simple brain mapping and brain activation identification during specific tasks and in response to specific events, and analyzed differences in brain activation depending on different conditions. Three studies (i.e., Castelhano et al. (2018), Kosti et al. (2018), Lee et al. (2016)) performed a more in-depth analysis and additionally looked into functional brain connectivity (i.e., the organization, inter-relationship, and integrated performance of different brain regions (Bastos and Schoffelen, 2016; Rogers et al., 2008)). NeuroSE researchers must keep in mind that the emergence of complex mental processes such as code comprehension is based on activity in a network of brain regions rather than on activity in one area alone; see, for example, Appendix C in Riedl et al. (2017a) which summarizes cognitive neuroscience work without using terminology specific to neuroscience. Accordingly, more sophisticated data analysis techniques are needed in future brain research studies which consider that simple one-to-one-mappings between cognition and brain areas do not exist (see, for example Hubert et al. (2017) who introduced some techniques recently).

In this context, Friston (1994) distinguishes four major concepts: functional specialization (i.e., analyses of region-specific effects: Which brain regions are involved in a specific mental process?), functional integration (i.e., analyses of effects between brain regions: Which interactions exist between brain regions so that a specific mental processes emerges?), functional connectivity (i.e., the temporal correlation between regionally separate brain processes), and effective connectivity (i.e., the influence of one brain region on another region). Note that functional and effective connectivity are subcategories of functional integration. First, current NeuroSE fMRI research focused on functional specialization. Second, in cognitive neuroscience functional integration and its subcategories have received much attention in the fMRI literature in the past two decades, based on seminal methods papers (e.g., Friston (2002), O'Reilly et al. (2012), Penny et al. (2004)). It follows that in cognitive neuroscience, if compared to current NeuroSE research, brain mechanisms underlying human cognition and emotion are often studied in a more realistic manner. Yet, because NeuroSE is a young discipline, this fact is more a call for corresponding future research rather than

a fundamental critique of the currently available research. Third, unlike functional connectivity analyses in fMRI research, network examinations in fNIRS have only recently begun to become popular (e.g., Li and Yu (2018)). However, because tool boxes are also increasingly available to study near-infrared spectroscopy data from a connectivity perspective (e.g., Xu et al. (2015)), we foresee more corresponding studies in the future. Note that in the current fNIRS literature, network studies predominantly refer to resting state situations. Here, connectivity between brain regions that occurs when an explicit task is *not* being performed is studied (e.g., Buckner et al. (2013), Sharaev et al. (2016)). Yet, in NeuroSE research a number of research contexts are imaginable in which this kind of network analyses may be useful. Imagine, for example, the study of functional resting state differences between highly and less experienced programmers. Moreover, in fNIRS optodes placement is critical for regions-of-interest analyses, and recently a toolbox for probe arrangement has been presented in the literature (Zimeo Morais et al., 2018).

Studies related to the ANS have mostly relied on data-driven approaches (in particular classification). In the future, due to advances in machine learning and the availability of low cost measurement instruments, we particularly expect an increase in such studies. The biggest challenge here is, however, that it is unclear to what extent the developed models can be generalized or need to be developed and trained for a specific context. These insights call for additional research to establish a better understanding to what extent classifiers for mental states trained on one software development activity can be used for another software development activity.

5.5. Validity threats

The following four major validity threats were identified and mitigated in relation to the review:

Descriptive Validity. Descriptive validity concerns the extent to which observations are described accurately and objectively. To mitigate this threat and to objectify the data collection process we designed a data collection form that we could always revisit.

Theoretical Validity. Theoretical validity refers to the ability to capture what we intend to capture. To mitigate the risk that studies are missed we carefully designed the search string by systematically combining a "Neuro" term with a SE term and complemented the search with a backward and forward search. To reduce bias in data extraction and classification the first author and second author extracted the data and if borderline cases appeared, they were discussed by the first author and the second author, with the involvement of the third author.

Interpretive validity. Interpretive validity is ensured when the conclusions are drawn from the data. A threat to interpretative validity is researcher bias. None of the authors is a coauthor of a paper, which was reviewed. Moreover, collectively the coauthors of this review have approximately three decades of experience in neurophysiological measurements that may help in the interpretation of data.

Repeatability. Repeatability requires a detailed description of the research process including data analysis. We explained in detail the process we followed. Moreover, we followed existing guidelines on how to conduct SLRs and have made the data extraction form and a file with all study details available as supplementary material.

6. Conclusion and future work

This paper maps the literature using measurements of brain and autonomic nervous system activity in a software engineering context and provides a comprehensive overview of the NeuroSE literature. We hope that this literature review will make it easier for other researchers to engage in NeuroSE research. To lower the entry barriers for conducting NeuroSE research, *methodological contributions* providing guidelines on how to conduct NeuroSE research would be helpful. Moreover, we also regard *infrastructural contributions* that support the collection and analysis of neurophysiological data in an SE context as critical (e.g., tools for linking of neurophysiological measures with the software artifacts in the vein of iTrace (Guarnera et al., 2018) or tools for the multimodal exploration of data (Peitek et al., 2019; Roy et al., 2020)). Both methodological and infrastructural contributions can play an important role in helping to increase the number of studies collecting neurophysiological data in the future.

Overall, neurophysiological measurements can contribute to the field of software engineering in several ways: (1) Neurophysiological measurements can contribute to enhancing our understanding of *human factors in software engineering*. For example, neurophysiological measurements can be used to investigate cognitive as well as emotional processes of developers and hence can complement subjective (self-reported) and behavioral measurements. (2) Studies using neurophysiological measurements can inform the *development of methods, tools, and techniques to improve the development of software.* (3) Neurophysiological measurements can play an important role in understanding the use of software systems, i.e., *software analytics*, and provide insights into how software is perceived by their users. (4) Neurophysiological measurements can be used to develop so-called *neuro-adaptive systems*, software systems that are able to adapt to the mental state of their users (based on neurophysiological information, which constitutes a correlate of that mental state). This can include neuro-adaptive integrated development environments, but also neuro-adaptive learning platforms (e.g., for programming). While the NeuroSE field is still at a nascent stage, we conclude that what has been revealed so far constitutes a valuable basis for future research. We believe that a better understanding of brain processes and processes related to ANS activity will contribute to advancements in software engineering and better software systems.

CRediT authorship contribution statement

Barbara Weber: Conceptualization, Validation, Investigation, Formal analysis, Writing - original draft. **Thomas Fischer:** Validation, Investigation, Formal analysis, Writing - review & editing. **René Riedl:** Conceptualization, Methodology, Validation, Writing - original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Overview of selected empirical papers

See Table 10.

Table 10
Selected empirical papers (N=47).

ID	Year	Authors	Outlet	Type of Outlet	Title
A024	2014	Fritz, Begel, Müller, Yigit-Elliott, Züger	International Conference on Software Engineering (ICSE)	Conference	Using psycho-physiological measures to assess task difficulty in software development
A028	2014	Ikutani and Uwano	International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing	Conference	Brain activity measurement during program comprehension with NIRS
A029	2014	Nakagawa, Kamei, Uwano, Monden, Matsumot, German	International Conference on Software Engineering (ICSE)	Conference	Quantifying programmers' mental workload during program comprehension based on cerebral blood flow measurement: A controlled experiment
A030	2014	Siegmund, Kästner, Apel, Parnin, Bethmann, Leich, Saake, Brechmann	International Conference on Software Engineering (ICSE)	Conference	Understanding source code with functional magnetic resonance imaging
A034	2015	González, Robbes, Góngora, Medina	International Conference on Augmented Cognition (ICAC)	Conference	Measuring Concentration While Programming with Low-Cost BCI Devices: Differences Between Debugging and Creativity Tasks
A036	2015	Jbara and Feitelson	International Conference on Program Comprehension (ICPC)	Conference	How Programmers Read Regular Code: A Controlled Experiment Using Eye Tracking
A040	2015	Müller and Fritz	International Conference on Mobile Software Engineering and Systems (ICMSES)	Conference	Stuck and Frustrated or in Flow and Happy: Sensing Developers' Emotions and Progress
A048	2016	Crk, Kluthe, Stefik	ACM Transactions on Computer-Human Interaction	Journal	Understanding Programming Expertise: An Empirical Study of Phasic Brain Wave Changes
A049	2016	Duraes, Madeira, Castelhano, Duarte, Castelo-Branco	International Symposium on Software Reliability Engineering (ISSRE)	Conference	WAP: Understanding the Brain at Software Debugging
A055	2016	Müller and Fritz	International Conference on Software Engineering (ICSE)	Conference	Using (bio)metrics to predict code quality online
A069	2017	Floyd, Santander, Weimer	International Conference on Software Engineering (ICSE)	Conference	Decoding the Representation of Code in the Brain: An fMRI Study of Code Review and Expertise
A077	2017	Siegmund, Peitek, Parnin, Apel, Hofmeister, Kästner, Begel	International Symposium on Foundations of Software Engineering (FSE)	Conference	Measuring neural efficiency of program comprehension
A078	2017	Yeh, Gopstein, DanYan, Zhuang	Frontiers in Education Conference	Conference	Detecting and comparing brain activity in short program comprehension using EEG
A083 and A132	2018	Fakhoury, Ma, Arnaoudova, Adesope	International Conference on Program Comprehension (ICPC)	Conference	The effect of poor source code lexicon and readability on developers' cognitive load
	2020	Fakhoury, Roy, Ma, Arnadoudova, Adesope	Empirical Software Engineering	Journal	Measuring the impact of lexical and structural inconsistencies on developers' cognitive load during bug localization
A087	2018	Peitek, Siegmund, Apel, Kästner, Parnin, Bethmann, Leich, Saake, Brechmann	IEEE Transactions on Software Engineering	Journal	A Look into Programmers' Heads
B010	2006	Aschwanden and Crosby	HICSS	Conference	Code Scanning Patterns in Program Comprehension
B027	2014	Crk and Kluthe	International Conference on Engineering in Medicine and Biology Society (EMBC)	Conference	Toward using alpha and theta brain waves to quantify programmer expertise
B029	2015	Züger and Fritz	Conference on Human Factors in Computing Systems (CHI)	Conference	Interruptibility of Software Developers and its Prediction Using Psycho-Physiological Sensors

(*continued on next page*)

Table 10 (*continued*).

ID	Year	Authors	Outlet	Type of Outlet	Title
B030	2016	Lee, Matteson, Hooshyar, Kim, Jung, Nam, Lim	International Conference on Bioinformatics and Bioengineering	Conference	Comparing Programming Language Comprehension between Novice and Expert Programmers using EEG Analysis
B032	2018	Castelhano, Duarte, Ferreira, Duraes, Madeira, Castelo-Branco	Brain Imaging and Behavior	Journal	The role of the insula in intuitive expert bug detection in computer code: an fMRI study
F023	2016	Crk and Kluthe	International Conference on Engineering in Medicine and Biology Society (EMBC)	Conference	Leveraging Biometric Data to Boost Software Developer Productivity
F026	2016	Yamamoto, Uwano, Ikutani	International Conference on Applied Computing and Information Technology/ Computational Science/Intelligence and Applied Informatics/Big Data, Cloud Computing, Data Science & Engineering	Conference	Programmer's Electroencephalogram Who Found Implementation Strategy
F028	2017	Duraisingam, Palaniappan, Andrews	Conference on Emerging Devices and Smart Systems	Conference	Cognitive task difficulty analysis using EEG and data mining
F029	2017	Lee, Hooshyar, Ji, Nam, Lim	Cluster Computing	Journal	Mining biometric data to predict programmer expertise and task difficulty
F031	2018	Kosti, Georgiadis, Adamos, Laskaris, Spinellis, Angelis	International Journal of Human-Computer Studies	Journal	Towards an affordable brain computer interface for the assessment of programmers' mental workload
F038	2018	Züger, Müller, Meyer, Fritz	Conference on Human Factors in Computing Systems (CHI)	Conference	Sensing Interruptibility in the Office: A Field Study on the Use of Biometric and Computer Interaction Sensors
A096	2018	Behroozi and Parnin	Workshop on Eye Movements in Programming (EMIP)	Conference	Can we predict stressful technical interview settings through eye-tracking?
A100	2019	Ahrens, Schneider and Busch	Workshop on Eye Movements in Programming (EMIP)	Conference	Attention in Software Maintenance: An Eye Tracking Study
A102	2019	Couceiro, Duarte, Duraes, Castelhano, Duarte, Teixeira, Castelo, Carvalho, Madeira	International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)	Conference	Biofeedback Augmented Software Engineering: Monitoring of Programmers' Mental Effort
A103	2019	Couceiro, Carvalho, Castelo, Madeira, Barbosa, Duraes, Duarte, Castelhano, Duarte, Teixeira, Laranjeiro, Medeiros	International Symposium on Software Reliability Engineering (ISSRE)	Conference	Spotting Problematic Code Lines using Nonintrusive Programmers' Biofeedback
A104	2019	Couceiro, Duarte, Duraes, Castelhano, Duarte, Teixeira, Castelo, Carvalho, Madeira	International Conference on Dependable Systems and Networks (DSN)	Conference	Pupillography as Indicator of Programmers' Mental Effort and Cognitive Overload
A106	2019	Fucci, Girardi, Novielli, Quaranta and Lanubile	International Conference on Program Comprehension (ICPC)	Conference	A Replication Study on Code Comprehension and Expertise using Lightweight Biometric Sensors
A110	2019	Huang, Liu, Krueger, Santander, Hu, Leach, Weimer	International Conference on Software Engineering (ICSE)	Conference	Distilling Neural Representations of Data Structure Manipulation using fMRI and fNIRS
A112	2019	Ishida and Uwano	International Companion Conference on Art, Science, and Engineering of Programming	Conference	Time series analysis of programmer's EEG for debug state classification
A113	2019	Ishida and Uwano	Workshop on Eye Movements in Programming (EMIP)	Conference	Synchronized Analysis of Eye Movement and EEG during Program Comprehension
A127	2019	Wulff-Jensen, Ruder, Triantafyllou and Bruni	International Conference on Neuroergonomics and Cognitive Engineering	Conference	Gaze Strategies Can Reveal the Impact of Source Code Features on the Cognitive Load of Novice Programmers

(*continued on next page*)

Table 10 (*continued*).

ID	Year	Authors	Outlet	Type of Outlet	Title
A138	2020	Vrzakova, Begel, Mehtätalo Bednarik	Journal of Systems and Software	Journal	Affect Recognition in Code Review: An In-situ Biometric Study of Reviewer's Affect
B037	2016	Ahonen, Cowley, Torniainen, Ukkonen, Vihavainen, Puolamäki	PLOS ONE	Journal	Cognitive Collaboration Found in Cardiac Physiology: Study in Classroom Environment
F040 and F047	2019	Doukakis, Spyridon	AIMS Neuroscience	Journal	Exploring brain activity and transforming knowledge in visual and textual programming using neuroeducation approaches
	2020	Doukakis, Papalaskari, Vlamos, Plerou, Giannopoulou	World Congress on Genetics, Geriatrics, and Neuro-degenerative Disease Research (GeNeDis)	Conference	Undergraduate Students' Brain Activity in Visual and Textual Programming
F041	2019	Ikramov, Ivanov, Masyagin, Shakirov, Sirazidtinov, Succi, Thapaliya, Tormasov, Zufarova	International Conference on Software Engineering and Knowledge Engineering	Conference	Initial evaluation of the brain activity under different software development situations
F042	2019	Medeiros, Couceiro, Castelhano Castelo Branco, Duarte, Duarte, Duraes, Madeira, Carvalho, Teixeira	International Conference of the Engineering in Medicine and Biology Society (EMBC)	Conference	Software code complexity assessment using EEG features
F048	2020	Girardi, Novielli, Fucci, Lanubile	International Conference on Software Engineering (ICSE)	–	Recognizing Developers' Emotions while Programming
F049	2020	Ikutani, Kubo, Nishida, Hata, Matsumoto, Ikeda, Nishimoto	eNeuro	–	Expert programmers have fine-tuned cortical representations of source code
F050	2020	Ivanova, Srikant, Sueoka, Kean, Dhamala, O'Reilly, Bers, Fedorenko	eLife	–	Comprehension of computer code relies primarily on domain-general executive resources
F051	2020	Liu, Kim, Wilson, Bedny	eLife	Journal-	Computer code comprehension shares neural resources with formal logical inference in the fronto-parietal network
B035	2018	Ahonen, Cowley, Hellas, Puolamäki	Scientific Reports	Journal	Biosignals reflect pair-dynamics in collaborative work: EDA and ECG study of pair-programming in a classroom environment
B036	2018	Behroozi, Lui, Moore, Ford, Parnin	International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)	Conference	Dazed: Measuring the Cognitive Load of Solving Technical Interview Problems at the Whiteboard

Appendix B. Overview of remaining selected papers

See Table 11.

Appendix C. Overview of types of contributions

See Table 12.

Appendix D. Usage of neurophysiological methods

See Table 13.

Appendix E. Overview of used measurement instruments

See Table 14.

Appendix F. Measures used as features for machine learning classifiers

See Table 15.

Appendix G. Research questions categorized by methods

See Table 16.

Appendix H. Overview of classifier performance

See Table 17.

Appendix I. Summary of study participant information

See Table 18.

Appendix J. Overview of used stimuli

See Table 19.

Appendix K. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jss.2021.110946.

Table 11
Remaining selected papers (N=42).

ID	Year	Authors	Outlet	Type of outlet	Title
A018	2011	Parnin	International Conference on Program Comprehension (ICPC)	Conference	Subvocalization - Toward Hearing the Inner Thoughts of Developers
A039	2015	Müller	International Conference on Mobile Software Engineering and Systems (ICMSES)	Conference	Measuring Software Developers' Perceived Difficulty with Biometric Sensors
A041	2015	Radevski, Hata, Matsumoto	International Workshop on Cooperative and Human Aspects of Software Engineering	Conference	Real-Time Monitoring of Neural State in Assessing and Improving Software Developers' Productivity
A043	2015	Shaffer, Wise, Walters, Müller, Falcone, Sharif	International Symposium on Foundations of Software Engineering (FSE)	Conference	iTrace: Enabling eye tracking on software artifacts within the IDE to support software engineering tasks
A044	2015	Sharafi, Shaffer, Sharif, Gueheneuc	Asia-Pacific Software Engineering Conference	Conference	Eye-Tracking Metrics in Software Engineering
A045	2015	Sharafi, Soh, Gueheneuc	Information and Software Technology	Journal	A systematic literature review on the usage of eye-tracking in software engineering
A060	2016	Sharif, Clark, Maletic	International Symposium on Foundations of Software Engineering (FSE)	Conference	Studying developer gaze to empower software engineering research and practice
A063	2016	Wise, Prox, Clark, Sharif	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Invited Talk: Towards an emotionally aware development environment
A070	2017	Fountaine, Sharif	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Emotional Awareness in Software Development: Theory and Measurement
A074	2017	Mi, Keung, Huang, Xiao	International Conference on Software Quality, Reliability and Security	Conference	Using Eye Tracking Technology to Analyze the Impact of Stylistic Inconsistency on Code Readability
A075	2017	Ostberg, Graziotin, Wagner, Derntl	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Towards the Assessment of Stress and Emotional Responses of a Salutogenesis-Enhanced Software Tool Using Psychophysiological Measurements
A080	2018	Busechian, Ivanov, Rogers, Sirazit, Succi, Tormasov, Yi	International Conference on Software Engineering (ICSE)	Conference	Understanding the impact of pair programming on the minds of developers
A086	2018	Peitek	International Conference on Software Engineering (ICSE)	Conference	A neuro-cognitive perspective of program comprehension
A088	2018	Peitek, Siegmund, Parnin, Apel, Hofmeister, Brechmann	International Symposium on Empirical Software Engineering and Measurement	Conference	Simultaneous measurement of program comprehension with fMRI and eye tracking
B018	2010	Parnin	Workshop of the Psychology of Programming Interest Group	Conference	A Cognitive Neuroscience Perspective on Memory for Programming Tasks
B025	2012	Siegmund, Brechmann, Apel, Kästner, Liebig, Leich, Saak	International Symposium on Foundations of Software Engineering (FSE)	Conference	Toward measuring program comprehension with functional magnetic resonance imaging
B031	2017	Peitek, Siegmund, Brechmann	Workshop on Eye Movements in Programming (EMIP)	Conference	Enhancing fMRI Studies of Program Comprehension with Eye-Tracking
B033	2018	Peitek, Siegmund, Parnin, Apel, Brechmann	Workshop on Eye Movements in Programming (EMIP)	Conference	Toward conjoint analysis of simultaneous eye-tracking and fMRI data for program-comprehension studies
F020	2015	Kontogiorgos, Manikas	Workshop on Eye Movements in Programming (EMIP)	Conference	Towards identifying programming expertise with the use of physiological measures

(*continued on next page*)

Table 11 (*continued*).

ID	Year	Authors	Outlet	Type of outlet	Title
F022	2016	Begel	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Invited Talk: Fun with Software Developers and Biometrics
F024	2016	Fritz, Müller	International Conference on Software Analysis, Evolution, and Reengineering	Conference	Leveraging Biometric Data to Boost Software Developer Productivity
F032	2018	Wrobel	Applied Sciences	Journal	Applicability of Emotion Recognition and Induction Methods to Study the Behavior of Programmers
F033	2018	Brown, Ivanov, Rogers, Succi, Tormasov, Yi	International Conference on Evaluation of Novel Approaches to Software Engineering	Conference	Toward a Better Understanding of How to Develop Software Under Stress – Drafting the Lines for Future Research
F034	2018	Fakhoury	European Software Engineering Conference	Conference	Moving Towards Objective Measures of Program Comprehension
F035	2018	Girardi, Lanubile, Novielli, Fucci	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Sensing developers' emotions: The Design of a Replicated Experiment
F036	2018	Peitek, Siegmund, Parnin, Apel, Brechmann	Workshop on Eye Movements in Programming (EMIP)	Conference	Beyond gaze: preliminary analysis of pupil dilation and blink rates in an fMRI study of program comprehension
F037	2018	Züger, Fritz	International Conference on Program Comprehension (ICPC)	Conference	Sensing and supporting software developers' focus
F052	2020	Siegmund, Peitek, Brechmann, Parnin, Apel	Communications of the ACM	Journal	Studying programming in the neuroage: Just a crazy idea?
A091	2016	Ford, Barik, abd Parnin	Workshop on Eye Movements in Programming (EMIP)	Conference	Studying Sustained Attention and Cognitive States with Eye Tracking in Remote Technical Interviews
A092	2016	Moffat and Paterson	Workshop on Eye Movements in Programming (EMIP)	Conference	Eye-tracking to trace anxieties of programmers
A093	2016	Nolan, Mooney, and Bergin	Workshop on Eye Movements in Programming (EMIP)	Conference	Examining the role of cognitive load when learning to program.
A107	2019	Girardi, Lanubile, Novielli, Quaranta, Serebrernik	Workshop on Emotion Awareness in Software Engineering (SEmotion)	Conference	Towards Recognizing the Emotions of Developers Using Biometrics: The Design of a Field Study
B034	2017	Minas, Kazman, Tempero	International Conference on Augmented Cognition (ICAC)	Conference	Neurophysiological Impact of Software Design Processes on Software Developers
F039	2018	Peitek, Siegmund, Parnin, Apel, Hofmeister, Kästner, Begel, Bethmann, Brechmann	Software Engineering and Software Management 2018	Conference	Neural Efficiency of Top-Down Program Comprehension
F044	2019	Tarasau, Thapaliya, Zufarova	International Conference on Software Technology: Methods and Tools	Conference	Problems in Experiment with Biological Signals in Software Engineering: The Case of the EEG
F045	2019	Thapaliya	International Young Scientists Conference on Information Technologies, Telecommunications and Control Systems (ITTCS 2019)	Conference	EEG: identification of concentration level under pair programming
F046	2020	Brown, Ivanov, Rogers, Succi, Tormasov, Yi	arXiv	–	Toward a Better Understanding of How to Develop Software Under Stress - Drafting the Lines for Future Research
A137	2020	Sharafi, Sharif, Guéhéneuc Begel, Bednarik, Crosby	Empirical Software Engineering	Journal	A practical guide on conducting eye tracking studies in software engineering
F043	2019	Peitek, Apel, Brechmann, Parnin, Siegmund	International Conference on Program Comprehension (ICPC)	Conference	CodersMUSE: Multi-Modal Data Exploration of Program-Comprehension Experiments

(*continued on next page*)

Table 11 (*continued*).

ID	Year	Authors	Outlet	Type of outlet	Title
A108	2019	Goncales, Farias, da Silva, Fessler	International Conference on Program Comprehension (ICPC)	Conference	Measuring the Cognitive Load of Software Developers: A Systematic Mapping Study
A122	2019	Sánchez-Gordón and Colomo-Palacio	Information and Software Technology	Journal	Taking the emotional pulse of software engineering — A systematic literature review of empirical studies
A134	2020	Menzen, Farias, and Bischoff	Behaviour & Information Technology	Journal	Using biometric data in software engineering: A systematic mapping study

Table 12
Types of contributions of NeuroSE research (N=89).

Type	Total	Papers
Empirical	47	(Ahonen et al., 2018, 2016; Ahrens et al., 2019; Aschwanden and Crosby, 2006; Behroozi et al., 2018; Behroozi and Parnin, 2018; Castelhano et al., 2018; Couceiro et al., 2019a,c,b; Crk et al., 2016; Crk and Kluthe, 2016, 2014; Doukakis, 2019; Doukakis et al., 2020; Duraes et al., 2016; Duraisingam et al., 2017; Fakhoury et al., 2020, 2018; Floyd et al., 2017; Fritz et al., 2014; Fucci et al., 2019; Girardi et al., 2020; González et al., 2015; Huang et al., 2019; Ikramov et al., 2019; Ikutani et al., 2020; Ikutani and Uwano, 2014; Ishida and Uwano, 2019b,a; Ivanova et al., 2020; Jbara and Feitelson, 2015; Kosti et al., 2018; Lee et al., 2017, 2016; Liu et al., 2020; Medeiros et al., 2019; Müller and Fritz, 2016, 2015; Nakagawa et al., 2014; Peitek et al., 2018a; Siegmund et al., 2017, 2014; Vrzakova et al., 2020; Wulff-Jensen et al., 2019; Yamamoto et al., 2016; Yeh et al., 2017; Züger and Fritz, 2018, 2015)
Empirical (research in progress)	24	(Brown et al., 2020; Busechian et al., 2018; Fakhoury, 2018; Ford et al., 2016; Fountaine and Sharif, 2017; Fucci et al., 2019; Girardi et al., 2019, 2018; Kontogiorgos and Manikas, 2015; Mi et al., 2017; Minas et al., 2017; Moffat and Paterson, 2016; Müller, 2015; Nolan et al., 2016; Ostberg et al., 2017; Parnin, 2011; Peitek, 2018; Peitek et al., 2018b,c,e, 2017; Radevski et al., 2015; Siegmund et al., 2012; Tarasau et al., 2019; Thapaliya, 2019)
Methodological	8	(Peitek et al., 2019, 2018d; Shaffer et al., 2015; Sharafi et al., 2020, 2015a; Sharif et al., 2016; Wise et al., 2016; Wrobel, 2018)
Conceptual	5	(Begel, 2016; Brown et al., 2018; Fritz and Muller, 2016; Parnin, 2010; Siegmund et al., 2020)
Review	5	(Goncales et al., 2019; Menzen et al., 2020; Sánchez-Gordón and Colomo-Palacios, 2019; Sharafi et al., 2015b; Züger and Fritz, 2018)

Table 13
Methods used in completed empirical NeuroSE research (N=47).

	Brain activity			Autonomic nervous system activity				Number methods used
	EEG	fMRI	fNIRS	Eye tracking	Skin	HR	Other	
Ahonen et al. (2016)						x		1
Ahonen et al. (2018)					x	x		2
Ahrens et al. (2019)				x				1
Aschwanden and Crosby (2006)				x				1
Behroozi and Parnin (2018)				x				1
Behroozi et al. (2018)				x				1
Castelhano et al. (2018)		x						1
Couceiro et al. (2019a)				x		x		2
Couceiro et al. (2019b)				x				1
Couceiro et al. (2019c)						x		1
Crk and Kluthe (2014)	x							1
Crk and Kluthe (2016)	x							1
Crk et al. (2016)	x							1
Doukakis (2019), Doukakis et al. (2020)	x							1
Duraes et al. (2016)		x						1
Duraisingam et al. (2017)	x							1
Fakhoury et al. (2020, 2018)			x					1
Floyd et al. (2017)		x						1
Fritz et al. (2014)	x			x	x			3
Fucci et al. (2019)	x				x	x		3
Girardi et al. (2020)	x				x	x		3
González et al. (2015)	x							1
Huang et al. (2019)		x	x					2
Ikramov et al. (2019)	x							1
Ikutani and Uwano (2014)			x					1

(*continued on next page*)

Table 13 (*continued*).

	Brain activity			Autonomic nervous system activity				Number methods used
	EEG	fMRI	fNIRS	Eye tracking	Skin	HR	Other	
Ikutani et al. (2020)		x						1
Ishida and Uwano (2019a)	x							1
Ishida and Uwano (2019b)	x							1
Ivanova et al. (2020)		x						1
Jbara and Feitelson (2015)				x				1
Kosti et al. (2018)	x							1
Lee et al. (2016)	x							1
Lee et al. (2017)	x			(x)[a]				1
Liu et al. (2020)		x						1
Medeiros et al. (2019)	x							1
Müller and Fritz (2015)					x	x	x	3
Müller and Fritz (2016)	x			x	x	x		4
Nakagawa et al. (2014)			x					1
Peitek et al. (2018a)		x						1
Siegmund et al. (2014)		x						1
Siegmund et al. (2017)		x						1
Vrzakova et al. (2020)					x			1
Wulff-Jensen et al. (2019)				x				1
Yamamoto et al. (2016)	x							1
Yeh et al. (2017)	x							1
Züger and Fritz (2015)	x				x	x		3
Züger et al. (2018)						x		1
Total	**20**	**10**	**4**	**10**	**8**	**10**	**1**	

[a](Lee et al., 2017) uses eye tracking, but the eye tracking measures used are not detailed. Thus, it is unclear whether the paper uses neurophysiological measures like pupil size or eye blinking rate.

Table 14
Measurement instruments used in completed empirical NeuroSE research (N=47).

Measurement Instrument	Method	Total	Papers
Emotiv EPOC, 14 channels	EEG	5	(Crk et al., 2016; Crk and Kluthe, 2016, 2014; Duraisingam et al., 2017; Kosti et al., 2018)
NeuroSky MindSet, 1 channel	EEG	3	(Fritz et al., 2014; Müller and Fritz, 2015; Züger and Fritz, 2015)
NeXus 10 MARK II by Mind Media (up to 4 channels) (no information concerning sensors)	EEG	3	(Ishida and Uwano, 2019b,a; Yamamoto et al., 2016)
Mitsar SMART-BCI, 24 channels	EEG	1	(Ikramov et al., 2019)
actiCAP, 32 channels and 16-channel V-Amp amplifier (Brain Products)	EEG	2	(Lee et al., 2017, 2016)[a]
Emotiv EPOC+, 14 channels	EEG	1	(Yeh et al., 2017)
NeuroSky MindWave	EEG	1	(González et al., 2015)
BrainLink Pro	EEG	1	(Girardi et al., 2020)
Neuroscan cap device (Compumedics), 64 channels	EEG	1	(Medeiros et al., 2019)
BIOPAC MP150 (no information concerning sensors)	EEG	1	(Doukakis, 2019; Doukakis et al., 2020)[b]
3T Magnetom Trio Tim MRI scanner, Siemens	fMRI	7	(Castelhano et al., 2018; Duraes et al., 2016; Floyd et al., 2017; Ivanova et al., 2020; Peitek et al., 2018a; Siegmund et al., 2017, 2014)
3T Magnetom Prisma scanner by Siemens with a 64-channel head coil	fMRI	1	(Ikutani et al., 2020)
3T General Electric MR750 with a 32-channel head coil	fMRI	1	(Huang et al., 2019)
3T Phillips Achieva Multix X-Series scanner	fMRI	1	(Liu et al., 2020)
fNIR100 by BIOPAC	fNIRS	1	(Fakhoury et al., 2020, 2018)[b]
Nexus 10 with Nexus HEG sensor	fNIRS	1	(Ikutani and Uwano, 2014)
Wearable Hikari Topography WOT - 200 by Hitachi Medico	fNIRS	1	(Nakagawa et al., 2014)
CW6 fNIRS byTechEn Inc.; cap with 16 light emitters and 32 detectors	fNIRS	1	(Huang et al., 2019)
Eye Tribe 60 Hz	Eye tracking	3	(Jbara and Feitelson, 2015; Müller and Fritz, 2015; Wulff-Jensen et al., 2019)

(*continued on next page*)

Table 14 (*continued*).

Measurement Instrument	Method	Total	Papers
SMI eye tracking glasses 60 Hz	Eye tracking	2	(Behroozi et al., 2018; Behroozi and Parnin, 2018)
SMI eye tracker (no further information)	Eye tracking	2	(Couceiro et al., 2019a,b)
Tobii TX300	Eye tracking	1	(Fritz et al., 2014)
Tobii X3–120	Eye tracking	1	(Ahrens et al., 2019)
ASL eye tracking system	Eye tracking	1	(Aschwanden and Crosby, 2006)
Affective Q sensor	Skin	1	(Fritz et al., 2014)
Shimmer3 GSR+ (Realtime Technologies Ltd)	Skin	2	(Ahonen et al., 2018; Vrzakova et al., 2020)
Empatica E4 wristband	Skin, Heart (PPG)	3	(Fucci et al., 2019; Girardi et al., 2020; Müller and Fritz, 2016)
Empatica E3 wristband	Skin, Heart (PPG)	2	(Müller and Fritz, 2015; Züger and Fritz, 2015)
SenseCore chest strap	Skin (except EDA), Heart (ECG), Breath	1	(Müller and Fritz, 2016)
BiosignalsPlus toolkit with ECG sensor	Heart (ECG)	2	(Couceiro et al., 2019c,a)
eMotion Faros 180° devices by Mega Electronics Ltd	Heart (ECG)	2	(Ahonen et al., 2018, 2016)
Fitbit Charge 2	Heart (ECG)	1	(Züger et al., 2018)
Polar H7 chest belt and wristwatch	Heart (ECG)	1	(Züger et al., 2018)

[a]Channel information only provided for (Lee et al., 2017), but not for (Lee et al., 2016)
[b]Note that these papers have been merged and are therefore only counted once.

Table 15
Overview of measures used as features for machine learning classifiers (N=19).

Variable	Measures	Usage
Cognitive load	**EEG**:	Lee et al. (2017)
	frequency bands	Duraisingam et al. (2017), Kosti et al. (2018) Medeiros et al. (2019)
	frequency ratios	Fritz et al. (2014)
	compound signals	Fritz et al. (2014)
	eye blink rate	Fritz et al. (2014)
	interhemispheric differences	Duraisingam et al. (2017)
	functional connectivity measures	Kosti et al. (2018)
	Heart:	
	heart rate	Müller and Fritz (2016)
	heart rate variability	Müller and Fritz (2016), Couceiro et al. (2019a)
	Skin:	
	EDA Signal	Fritz et al. (2014), Müller and Fritz (2016)
	skin temperature	Müller and Fritz (2016)
	Eye:	
	pupil size	Fritz et al. (2014)
	Breath:	
	respiratory rate	Müller and Fritz (2016)
	Additional measures:	
	eye gaze	Fritz et al. (2014), Lee et al. (2017)
	interaction, code, and change metrics	Müller and Fritz (2016)
Affective state	**EEG**:	
	frequency bands and frequency ratios	Müller and Fritz (2015), Girardi et al. (2020)
	compound signals	Müller and Fritz (2015), Girardi et al. (2020)
	eye blink rate	Müller and Fritz (2015)
	Heart:	
	heart rate	Müller and Fritz (2015), Girardi et al. (2020)
	heart rate variability	Müller and Fritz (2015)
	blood volume pulse	Müller and Fritz (2015), Girardi et al. (2020)
	Skin:	
	EDA Signal	Müller and Fritz (2015), Vrzakova et al. (2020)
	skin temperature	Müller and Fritz (2015)
	Eye:	
	pupil size	Müller and Fritz (2015)
	Additional measures:	
	eye gaze	Müller and Fritz (2015), Vrzakova et al. (2020)
	touch	Vrzakova et al. (2020)

(*continued on next page*)

Table 15 (*continued*).

Variable	Measures	Usage
Stress	**Eye**:	
	pupil size	Behroozi and Parnin (2018)
	Additional measures:	
	eye gaze (various fixation-based and saccadic-measures)	Behroozi and Parnin (2018)
Interruptibility	**EEG**:	
	frequency bands and frequency ratios	Züger and Fritz (2015)
	compound signals	Züger and Fritz (2015)
	eye blink rate	Züger and Fritz (2015)
	Heart:	
	heart rate	Züger and Fritz (2015), Züger et al. (2018)
	heart rate variability	Züger and Fritz (2015), Züger et al. (2018)
	blood volume pulse	Züger and Fritz (2015)
	Skin:	
	EDA Signal	Züger and Fritz (2015)
	skin temperature	Züger and Fritz (2015)
	Additional measures:	
	Computer interaction, sleep, and physical activity	Züger et al. (2018)
Presence of quality concerns	**Heart**:	
	heart rate	Müller and Fritz (2016)
	heart rate variability	Müller and Fritz (2016)
	Skin:	
	EDA Signal	Müller and Fritz (2016)
	skin temperature	Müller and Fritz (2016)
	Breath:	
	Respiratory rate	Müller and Fritz (2016)
	Additional measures:	
	interaction, code, and change metrics	Müller and Fritz (2016)
Expertise	**EEG**:	Lee et al. (2017)
	frequency bands	Crk and Kluthe (2014)
	Additional measures:	
	eye gaze	Lee et al. (2017)
Software development activity	**EEG**:	
	frequency bands and frequency ratios	Fucci et al. (2019)
	compound signals	Fucci et al. (2019)
	Heart:	
	heart rate and heart rate variability	Fucci et al. (2019)
	blood volume pulse	Fucci et al. (2019)
	Skin:	
	EDA Signal	Fucci et al. (2019)
	fMRI:	
	fMRI BOLD signal	Floyd et al. (2017)
Task (sub)category + Problem structure	**fMRI**:	
	fMRI BOLD signal	Ikutani et al. (2020), Liu et al. (2020)
Correctness of responses	**EEG**:	
	frequency bands	Crk and Kluthe (2016)

Table 16
Overview of research questions addressed by different neurophysiological methods (N=47).

Research questions answered using fMRI
• Which brain regions are activated during program comprehension? (Siegmund et al., 2014)
• Can we replicate the results of Siegmund et al. (2014)? (Siegmund et al., 2017)
• What is the difference between bottom-up program comprehension and comprehension with semantic cues in terms of activation and the brain areas involved? (Siegmund et al., 2017)
• How do layout and beacons in source code influence program comprehension? (Siegmund et al., 2017)
• Which brain regions are activated during bottom-up program comprehension? (Peitek et al., 2018a)
• Does source-code complexity correlate with concentration levels during bottom-up program comprehension? (Peitek et al., 2018a)
• Does programming experience correlate with brain activation strength during bottom-up program comprehension? (Peitek et al., 2018a)

(*continued on next page*)

Table 16 (*continued*).

• Are neural representations of programming languages and natural languages distinct? (Floyd et al., 2017) • Can we relate tasks to brain regions? (Floyd et al., 2017) • Can we relate expertise to classification accuracy? (Floyd et al., 2017) • What are the brain activation patterns associated with bug confirmation? (Duraes et al., 2016) • What are the brain activation patterns associated with bug suspicion (in contrast to bug confirmation)? (Duraes et al., 2016) • What are the brain activation patterns for software program processing in expert participants while performing bug detection tasks? (Castelhano et al., 2018) • What are the differences in brain activation patterns during "suspicion" vs. bug detection events? (Castelhano et al., 2018) • Does activation in the anterior insula correlate with bug detection precision? (Castelhano et al., 2018) • What are the regions that influence/are influenced by the insular Region of Interest (ROI)? (Castelhano et al., 2018) • How does brain activation evoked by code comprehension differ from brain activation evoked from the underlying problem content? (Ivanova et al., 2020) • Are there brain regions in the multiple demand (MD) system that are functionally specialized for code comprehension? (Ivanova et al., 2020) • What is the role of the language system during code comprehension? (Ivanova et al., 2020) • What are code sensitive brain regions outside the MD/language system? (Ivanova et al., 2020) • What is the role of programming expertise on activation in the MD/language system? (Ivanova et al., 2020) • Does code comprehension have a consistent neural signature across people? Liu et al. (2020) • Is the neural signature of code comprehension similar to other culturally derived symbol systems (i.e., logic and math) or similar to natural language? Liu et al. (2020) • Can functional categories of source code be decoded from a developer's brain activity? (Ikutani et al., 2020)
Research questions answered using fNIRS
• What are the effects of variables and control statements on brain activation during program comprehension? (Ikutani and Uwano, 2014) • Can the measurement of brain blood-flow quantify programmers' mental workload during program comprehension activities? (Nakagawa et al., 2014) • Can developers' cognitive load be accurately associated with identifiers' terms using fNIRS and eye tracking devices? (Fakhoury et al., 2020, 2018) • Do inconsistencies in the source code lexicon cause a measurable increase in developers' cognitive load during program comprehension? (Fakhoury et al., 2020, 2018) • Do structural inconsistencies related to the readability of the source code cause a measurable increase in developers' cognitive load during program comprehension? (Fakhoury et al., 2020, 2018) • Do both structural and lexical inconsistencies combined cause a measurable increase in developers' cognitive load during program comprehension? (Fakhoury et al., 2020, 2018) • Does the presence of inconsistencies in the source code lexicon affect the cognitive load of developers (measured using fNIRS) over an entire source code snippet or only over the identifiers that are involved in the inconsistencies? (Fakhoury et al., 2020) • Are self-reported measures consistent with cognitive load (measured using fNIRS) and fixation duration data? (Fakhoury et al., 2020)
Research questions answered using fMRI and fNIRS
• Do data structure manipulations involve spatial ability? (Huang et al., 2019) • What is the impact of task difficulty on brain activation? (Huang et al., 2019) • Do fMRI and fNIRS agree for software engineering? (Huang et al., 2019) • How do self-reporting and neuroimaging compare? (Huang et al., 2019)
Research questions answered using EEG
• How do comprehension and syntax tasks differ in terms of patterns of brain activation? (Kosti et al., 2018) • Can we identify neural correlates of programmer's workload? (Kosti et al., 2018) • Can we create a model of subjective difficulty based on the recorded brainwave patterns? (Kosti et al., 2018) • Is there a difference in the brain activation depending on the software development activity (for different interaction types)? (González et al., 2015) • Is there a difference in the brain activation depending on the task difficulty (confusing versus non-confusing)? (Yeh et al., 2017) • Can a subject's performance (in terms of correct answers) be predicted based on broad alpha power? (Yeh et al., 2017) • Is task difficulty reflected in EEG electrical signal within programming comprehension tasks? (Duraisingam et al., 2017) • Can task difficulty (specifically program comprehension) be predicted accurately using a machine learning classifier with EEG features? (Duraisingam et al., 2017) • Can EEG indicators of working memory during code comprehension tasks be used for evaluating expertise-related differences in subject performance? (Crk et al., 2016) • What differences exist in the information processing between novices and experts during program comprehension? (Lee et al., 2016) • Can EEG be used as an index for classification of programmers who fail to find an implementation strategy? (Yamamoto et al., 2016) • Can EEG data be used to accurately predict the expertise of a developer during code comprehension tasks? (Crk and Kluthe, 2014) • Can EEG data be used as a predictor of task performance during code comprehension tasks? (Crk and Kluthe, 2016) • Can task difficulty (easy/difficult) and programmer expertise (novice/expert) be predicted accurately using a machine learning classifier with EEG and eye tracking features? (Lee et al., 2017)[a] • Are there differences in the EEG signal at completion of the understanding step depending on task performance? (Ishida and Uwano, 2019b) • Are there differences in the EEG signal at completion of the judgment step depending on task performance? (Ishida and Uwano, 2019b) • Are there differences in time series changes depending on task performance during the understanding/bug-judgment step? (Ishida and Uwano, 2019b) • Are changes in the EEG signal and eye movement data during program comprehension synchronized with any comprehension process? (Ishida and Uwano, 2019a) • Are there differences in time series changes depending on task performance during the understanding tasks? (Ishida and Uwano, 2019a) • What is the impact of the programming role (solo, navigator, driver) on brain activation? (Ikramov et al., 2019) • What is the impact of music on brain activation? (Ikramov et al., 2019) • Which EEG features are best suited for building a classifier for code complexity? (Medeiros et al., 2019)

(*continued on next page*)

Table 16 (*continued*).

• Can we build a classifier for code complexity using EEG? (Medeiros et al., 2019) Are there differences in brain activation dependent on the programming language? (Doukakis, 2019; Doukakis et al., 2020)
Research questions answered using eye tracking
• What are the scanning patterns during program comprehension? (Aschwanden and Crosby, 2006) • Are developers' visual efforts equally divided among regular segments? (Jbara and Feitelson, 2015) • Does attention data representation have any effect on the efficiency and quality of software maintenance tasks? (Ahrens et al., 2019) • How do structural and textual readability features affect program comprehension? (Wulff-Jensen et al., 2019) • How much affect the characteristics of the interview setting eye tracking measurements? (Behroozi and Parnin, 2018) • Can we detect differences in stress and cognitive load between the paper and whiteboard technical interview settings? (Behroozi et al., 2018) • Can a developer's cognitive load be measured during code reading using pupillography? (Couceiro et al., 2019b) • Do measurements of cognitive load based on pupillography correlate with subjective measurements of cognitive load (NASA-TLX) (Couceiro et al., 2019b)
Research questions answered using heart-related measurements
• Can we build a classifier that predicts a software developer's interruptibility accurately in the field? (Züger et al., 2018) • Can a developer's cognitive load be measured during code reading using HRV? (Couceiro et al., 2019c) • Do HRV measurements of cognitive load correlate with subjective measurements of cognitive load? (Couceiro et al., 2019c) • Do source code complexity metrics correlate with subjective measurements of cognitive load? (Couceiro et al., 2019c) • Can we extract Social Physiological Compliance (SPC) from an ECG signal in a natural protocol? (Ahonen et al., 2016)
Research questions answered using multiple modalities
Eye tracking, Heart
• Can eye tracking together with HRV and pupillography be used to identify non intrusively code lines (and even lexical tokens inside code lines) that correspond to mental effort peaks? (Couceiro et al., 2019a)
Skin, Heart
• Can we extract Social Physiological Compliance (SPC) from an ECG signal in a natural protocol? (Ahonen et al., 2018) • Can windowed heart-related measurements be substituted by fast biosignals for examining SPC in a natural protocol? (Ahonen et al., 2018) • Can the physiological signals with high temporal resolution, found to reflect SPC, be associated with task related emotional valence and engagement? (Ahonen et al., 2018)
EEG, Eye tracking, Skin
• Can we acquire psycho-physiological measures from eye tracking, EDA and EEG sensors to accurately predict task difficulty? (Fritz et al., 2014) • Which combination of psycho-physiological sensors and associated features best predicts task difficulty? (Fritz et al., 2014) • Can we use psycho-physiological measures to predict task difficulty as the developer is working? (Fritz et al., 2014)
EEG, Skin, Heart
• Can we predict the interruptibility of a knowledge worker in a real-world working context using a combination of psycho-physiological sensors? (Züger and Fritz, 2015) • Can we classify which task a participant is undertaking based on signals collected from lightweight biometric sensors? (Fucci et al., 2019) • Can we relate expertise to classification accuracy? (Fucci et al., 2019) • What is the minimal set of non-invasive biometric sensors to recognize developers' emotions? (Girardi et al., 2020)
Heart, Skin, Breath
• Can we use biometrics to identify places in the code that are perceived to be more difficult by developers? (Müller and Fritz, 2016) • Can we use biometrics to identify code quality concerns found through peer code reviews? (Müller and Fritz, 2016) • How do biometrics compare to more traditional metrics for predicting perceived difficulty and detecting quality concerns? (Müller and Fritz, 2016)
EEG, Eye tracking, Skin, Heart
• Can we use biometric sensors to determine developers' emotions and progress during change tasks? (Müller and Fritz, 2015)
Skin, Eye tracking, Touch
• How do nonverbal physiological signals predict components of affect after the code review task? (Vrzakova et al., 2020)

[a] Lee et al. (2017) uses eye tracking, but the eye tracking measures used are not detailed. Thus, it is unclear whether the paper uses neurophysiological measures like pupil size or eye blinking rate. We therefore classified this study as an EEG study only.

Table 17
Overview of performance of machine learning classifiers (N=19).

Variable	Usage in study	Features and Performance
Software Development Activity	**Code Comprehension versus Prose Comprehension** Fucci et al. (2019)	Features: EEG, Heart, Skin Best classifier: Heart Balanced accuracy: 87%
	Code Comprehension vs. Code Inspection **Code Comprehension vs. Prose Review** **Code Inspection vs. Prose Review** Floyd et al. (2017)	Features: fMRI Review vs. Prose (balanced accuracy 70.83%) Comprehension vs. Prose (balanced accuracy 79.17%) Review vs. Comprehension (balanced accuracy 61.74%)

(*continued on next page*)

Table 17 (*continued*).

Variable	Usage in study	Features and Performance
Task (sub)category	**Task classification (4 classes) Subcategory classification (11 classes)** Ikutani et al. (2020)	Features: fMRI Brain regions with significant above chance (25% and 9.72%) classification accuracies reported
Problem Structure	**For vs. if functions** Liu et al. (2020)	Features: fMRI lateral prefrontal cortex (accuracy 64.7%) intraparietal sulcus (accuracy 67.4%) posterior middle temporal gyrus (accuracy 68.4%) early visual cortex (accuracy 55.7%)
Expertise	**2 state classification (Novice vs. Expert)** Crk and Kluthe (2014)	Features: EEG Accuracy (all responses): Task 1: 55% for both Upper Alpha and Theta ERD Task 2: 59% for Upper Alpha and 63% for Theta ERD Task 3: 59% for Upper Alpha and 63% for Theta ERD Accuracy (only correct responses): Task 1: 56% for Upper Alpha and 63% for Theta ERD Task 2: 56% for Upper Alpha and 74% for Theta ERD Task 3: 67% for Upper Alpha and 63% for Theta ERD
	2 state classification (Novice vs. Expert) Lee et al. (2017)	Features: EEG and Eye Best classifier: EEG and Eye 97.7% precision and 96.4% recall
Presence of quality concerns	**2 state classification** Müller and Fritz (2016)	Features: Heart, Skin, Breath, interaction, code, and change metrics Best Classifier (within participants): Biometric features 13% precision, 38.6% recall on method level 22.0% precision, 40% recall at class level Best Classifier (across participants): All features 9.7% precision, 63.6% recall at method level 17.5% precision, 30.4% recall at class level
Perceived Progress	**2-state classification (high vs. low)** Müller and Fritz (2015)	Features: EEG, Heart, Skin, Eye Accuracy: 67.70%, Precision: 67.85%, Recall 68.29% Most predictive features: ΔAlpha, ΔBeta/Theta ΔMeanTempPeakAmpl, MaxPupilSize, ΔMeanPupilSize, ΔMeanSCL
Stress	**2-state classification (stressed vs. non stressed)** Behroozi and Parnin (2018)	Features: Eye (pupil and eye gaze) Accuracy: 0.88 (Setting and Stress Rating labeling), 0.77 (Setting labeling) and 0.78 (Stress Rating labeling) Fixation-based measures (including pupil size) are more distinctive than saccadic measures.
Cognitive Load	**2 state classification (easy/difficult)** Lee et al. (2017)	Features: EEG and Eye Best classifier: EEG and Eye 64.9% precision and 68.6% recall
	2 state classification (easy/difficult) Fritz et al. (2014)	Features: EEG, Skin, Eye Fusion Model (EEG, Skin, Eye) New participants: Precision: 64.99%; Recall: 64.58% New task for particular participant: Precision: 84.38%; Recall: 69.79% New participant–task pairs: Precision and recall: 67.71% Best classifier: New participants (Eye): Precision: 69.16%; Recall: 65.83% New task for particular participant (EEG, Skin, Eye): Precision: 84.38%; Recall: 69.79% New participant–task pairs (EDA, Eye): Precision: 68.27%; Recall: 68.27% Classifier (Online), Fusion Model (EEG, Skin, Eye): New participants, 55s window: Precision: 68.04%; Recall: 58.55% New task for particular participant, 30s: Precision: 80.68%; Recall: 64.01% New participant–task pairs, 55s: Precision: 96.74%; Recall: 63.73%

(*continued on next page*)

Table 17 (*continued*).

Variable	Usage in study	Features and Performance
		Best Classifier (Online), EDA and EEG New participant–task pairs, 55s: Precision: 100%; Recall: 66.13%
	2 state classification (easy/difficult) Duraisingam et al. (2017)	Features: EEG Precision: 76.55% and recall: 80.03% Best feature group: asymmetry ratio Precision: 76.9% and recall: 73.4%
	4 state classification Kosti et al. (2018)	Features: EEG PLV measurements are better compared to SP measurements. The best PLV model achieved a classification percentage of 55%.
	6 state classification (1 to 6) Müller and Fritz (2016)	Features: Heart, Skin, Breath, interaction/code/change metrics Best classifier (All feature groups, During change task) Cohen's Kappa: 38% (method level), 36% (class level) After change task Cohen's Kappa: 48% (method level), 30% (class level) Classifier using biometrics, During change task Cohen's Kappa: 37% (method level), 34% (class level) After change task Cohen's Kappa: 46% (method level), 26% (class level) Replication: Best classifier, (All feature groups, during change task) Cohen's Kappa: 29% (method level), 22% (class level) After change task Cohen's Kappa: 38% (class level)
	2 state classification Couceiro et al. (2019a)	Features: Heart Code vs. Rest: Sensitivity: 97 $\pm$ 6%; Specificity: 100%, Rest vs. Code1/Code2/Code3: Sensitivity and specificity are above 88%. C1 vs. C2, C1 vs. C3: Sensitivities and specificities above 81% C2 vs. C3: Sensitivity and specificity: 46 $\pm$ 38%
Cognitive Load	**2 state classification** Medeiros et al. (2019)	Features: EEG Accuracy: 70%$\pm$ 15.72% Recall: 92.50% (easy), 59.27% (medium), 58.33% (advanced) Recall: 94.22% (easy), 49.42% (medium), 52.87% (advanced) Features found in most folds: features related to gamma activity features related to the right parietal region
Affective State	**2-state classification (Positive vs. Negative valance)** Müller and Fritz (2015)	Features: EEG, Heart, Skin, Eye Accuracy: 71.36%, Precision: 64.32% , Recall 82.03% Features with the highest predictive power are: ΔAlpha, ΔBeta/Theta MinPupilSize, ΔMeanHR
	2-state classification (Positive vs. Negative valance) **2-state classification (Low vs. High Arousal)** Girardi et al. (2020)	Features: EEG, Heart Best classifier (EEG, Heart), hold out: Valence: accuracy of 72%; 65% accuracy for arousal Best classifier (only Empatica), hold out: Valence: accuracy of 71%; 65% accuracy for arousal Best classifier (EEG, Heart), leave one out: Valence: accuracy of 69%; 61% accuracy for arousal Best classifier (EEG), leave one out: Valence: accuracy of 71%; 63% accuracy for arousal
	2-state classification (Positive vs. Negative valance) **2-state classification (Low vs. High Arousal)** Vrzakova et al. (2020)	Features: Skin, Eye gaze, Touch Valence: Best classifier: fusion model: accuracy = 90.0% True positive rate: 95.7%; True negative rate: 69.5% (better predictions for positive valance) Arousal: Best classifier: fusion model: accuracy = 83.9% True positive rate: 85.3%; True negative rate: 82.3%

(*continued on next page*)

Table 17 (*continued*).

Variable	Usage in study	Features and Performance
		Best classifier using one modality (eye gazes) Valence: accuracy = 85.8%; Arousal: accuracy = 76.6%. Classification (after task vs. first 5 min) Valence: accuracy = 85.5%; Arousal: accuracy = 79.8
Interruptibility Interruptibility	**2 state classification, 5 state classification** Züger and Fritz (2015)	Features EEG, Heart, Skin: 2-states: Lab: 91.5% accuracy, Field: 78.6% accuracy 2-states: Lab: 43.9% accuracy, Field: 32.5%% accuracy
	2 state, 3 state, and 7 state classification Züger et al. (2018)	Features: Heart, Computer interaction, sleep, and physical activity All features (within participants): 2 states: Average accuracies 75.3% 3 states: Average accuracies 65.5% 7 states: Average accuracies 42.5% Classifier (computer interactions, within participants): 2 states: 74.8% Classifier (biometrics, within participants): 2 states: 68.3% Best classifier (across participants): 2 states: Average accuracies 69.8%
Correctness of responses	**2 state classification (correct/incorrect)**: Crk and Kluthe (2016)	Features EEG: accuracy 80.1%; sensitivity 83:8%; specificity 76:3%; positive predictive value 78:3%; negative predictive value 82:2%

Table 18
Study participant information for completed empirical research (N=47).

ID	Number participants	Male	Female	Age	Mean age	Study population	Background
Fritz et al. (2014)	15	14	1	27–60	41.6	Professional developers	At least 2 years programming experience; knowledge in C#
Ikutani and Uwano (2014)	11	11	0	–	–	Students	Undergraduate students who finished first programming lecture before experiment
Nakagawa et al. (2014)	10	10	0	22–26	–	Students	At least 3 years of C experience
Siegmund et al. (2014)	17	15	2	–	–	Students	Undergraduate programming and Java experience
González et al. (2015)	32	26	6	20–24	21.04	Students	At least basic programming courses (covering data structures and Java)
Jbara and Feitelson (2015)	20	17	3	–	24.8	Students (18) and Faculty (2)	3rd year students in computer science
Müller and Fritz (2015)	17	16	1	20–51	–	Students (11) and professional developer (6)	PhD students with a major in CS
Crk et al. (2016)	34	–	–	18–39	–	Students	–
Duraes et al. (2016)	13	13	0	–	28	Professional developers	–
Müller and Fritz (2016)	10	9	1	23–45		Professional developers	Average professional experience 10.2 years; total software development experience 3–22 years
	5	5	0	25–30		Professional developers	Average professional experience of 5.8 years
Floyd et al. (2017)	29	18	11	–	–	Students	At least basic programming expertise
Siegmund et al. (2017)	11	9	2	–	25.30	Students (8) and professional developer (3)	5 computer science students, 3 mathematics students
Yeh et al. (2017)	8	–	–	–	–	Students	–
Fakhoury et al. (2020, 2018)	25	–	–	–	–	Students	Graduate, undergraduate and PhD; at least one course in C++ or Java; experience 1-15 years, avg. 3.5 years

(*continued on next page*)

Table 18 (*continued*).

ID	Number participants	Male	Female	Age	Mean age	Study population	Background
Peitek et al. (2018a)	17	15	2	–	–	Students	Undergraduate level of programming experience
	11				25.30	Students	All familiar with Java or C at a medium level or higher
Aschwanden and Crosby (2006)	15	–	–	–	–	Students	Students in Computer Science
Crk and Kluthe (2014)	34	–	–	–	–	Students	–
Züger and Fritz (2015)	10	9	1	–	–	Students	Graduates with a major in CS; avg. 4.1 years of professional experience and 10.4 years development experience
	10	9	1	–	–	Professional developers	Average 8.5 years of professional experience and 12.7 years total development experience
Lee et al. (2016)	18	13	5	–	24	Students	CS majoring students; all have Java experience
Castelhano et al. (2018)	19	18	1	–	28	Professional developers	All participants hold a master and/or PhD in CS or equivalent and have at least 3 years of experience in C; strong experience as formal code inspectors
Crk and Kluthe (2016)	33	–	–	–	–	Students	Undergraduates computer science
Yamamoto et al. (2016)	17	–	–	16–20	–	Students	–
Duraisingam et al. (2017)	8	7	1	–	–	Students	Postgraduate students with at least 6 months Java experience
Lee et al. (2017)	38 Overall	27	11	18–42	–		
	18 Experts	14	4	28–42	–	Professional (8) developers and students (10)	PhD students with major in CS and > 6 years software development experience
	20 Novices	13	7	18–29	–		
Kosti et al. (2018)	10	8	2	25–37	–	Faculty and students	Experience in C
Züger et al. (2018)	13	12	1	–	32.4	Professional developers	Avg. professional experience 6.5 years; development experience 11.8 years
Ahonen et al. (2016)	38	16	22	–	–	Students	age ranges: 28 from 18–23 years, 9 from 24–29 and one was over 34
Ahonen et al. (2018)	28	20	18	18–41	–	Students	Freshmen programming students
Behroozi et al. (2018)	11	–	–	–	–	Students	Background in computer science (8 graduates and 3 undergraduates)
Behroozi and Parnin (2018)	11	–	–	–	–	Students	Background in computer science (8 graduates and 3 undergraduates)
Ahrens et al. (2019)	22	17	5	–	–	Professional developers (6) and students (16)	16 students (8 senior undergraduate or graduate; 8 PhD students), experience in Java development and reading unfamiliar source code
Couceiro et al. (2019a)	30	24	6	–	24.4	–	Java experience: 12 intermediate, 14 advanced, 4 experts
Couceiro et al. (2019b)	30	24	6	–	24.4	–	Java experience: 12 intermediate, 14 advanced, 4 experts
Couceiro et al. (2019c)	26	–	–	–	–	–	All experienced in Java; Screening process assigned them to intermediate, advanced, expert
Doukakis (2019), Doukakis et al. (2020)	8	–	–	–	–	Students	First year computer science
Fucci et al. (2019)	28	24	4	–	–	Students	Undergraduate students
Huang et al. (2019)	Overall 70	46	24	–	–	Students	Computing background
	fMRI: 30	16	14	–	–		23 Undergraduate 7 Graduate
	fNIRS: 40	30	10	–	–		31 Undergraduate 9 Graduate

(*continued on next page*)

Table 18 (*continued*).

ID	Number participants	Male	Female	Age	Mean age	Study population	Background
Ikramov et al. (2019)	10	–	–	23–30	–	Students	Graduates with working experience
	3 2 2						Undergraduates
Ishida and Uwano (2019b)	5	5	0	19–20	–	Students	Undergraduate students; At least basic course in Java programming
Ishida and Uwano (2019a)	5	5	0	19–20	–	Students	Undergraduate students; At least basic course in Java programming
Medeiros et al. (2019)	30	–	–	–	–	–	–
Wulff-Jensen et al. (2019)	21	16	5	–	26.42	Students	Average experience level was 2.35 on a scale from 0– 6.
Girardi et al. (2020)	27	23	4	–	–	Students	21 undergraduates, 5 graduates, 1 post-graduate; Have finished capstone project using Java
Ikutani et al. (2020)	30	28	2	20–24	22.3	–	10 top 20% rankers (Expert); 10 21%–50% rankers (Middle) in AtCoder. Ten control subjects (Novice) with max four years programming experience and no experience in competitive programming. At least 6 month Java experience.
Ivanova et al. (2020)	24	9	15	–	–	–	Experiment 1: average number of years spent programming was 6.3 Both: Experience in the used programming language
	19	7	12	–	–	–	
Liu et al. (2020)	15	12	3	20–38	27.4	–	At least 5 years of programming experience; at least 3 years of experience working with Python.
Vrzakova et al. (2020)	37	35	2	25–43	34	Professional developers	active reviewers with more than ten reviews completed in the past three months, and more than five code reviews pending

Table 19
Stimuli used by completed empirical research (N=47).

ID	Task characteristics	Programming language	Size
Fritz et al. (2014)	10 comprehension tasks (2 practice, 8 experimental)	C#	–
Ikutani and Uwano (2014)	Program tasks (with different task types) and arithmetic tasks (with three level of difficulty)	C	3–6 lines
Nakagawa et al. (2014)	Three comprehension tasks (with two levels of difficulty); one used for familiarization	C	17–32 lines
Siegmund et al. (2014)	12 comprehension tasks	Java	max 20 lines
González et al. (2015)	1 programming/documenting/bug fixing task (both physical and digital)	Java	–
Jbara and Feitelson (2015)	2 code comprehension tasks	C	26–53
Müller and Fritz (2015)	2 change tasks (Java program interacting with Stack Exchange API and adding new feature in JHotDraw)	Java	–
Crk et al. (2016)	3 code comprehension tasks	Java	Small programs
Duraes et al. (2016)	6 software programs (3 with faults, i.e. code inspection and 3 without, i.e. code comprehension)	C	20–37 lines of C code; 1–3 screens
Müller and Fritz (2016)	No specific task, developers recorded during 2-week period during general software development activities	–	–

(*continued on next page*)

Table 19 (*continued*).

ID	Task characteristics	Programming language	Size
Floyd et al. (2017)	Comprehension task (Java snippets), Code review task: (Pull requests) Prose review task: Text stimuli	Java (code comprehension) and C (code review task) and text stimuli (prose review task)	20 lines (code comprehension) Max 2 files edited and 10 changed lines (code review)
Siegmund et al. (2017)	Code comprehension tasks	Java	8–19 lines
Yeh et al. (2017)	Code comprehension tasks 6 code snippets (in confusing and non-confusing version)	C/C++	6 lines on average
Fakhoury et al. (2020, 2018)	1 code comprehension task (code versus prose) and 3 bug localization tasks (with and without linguistic antipatterns, with and without structural inconsistencies)	Java	30–40 lines
Peitek et al. (2018a)	12 comprehension tasks	Java	Max 18 lines
Aschwanden and Crosby (2006)	6 programs in 2 versions (recursive versus non-recursive)	Java	–
Crk and Kluthe (2014)	3 code comprehension tasks	Java	Small programs
Züger and Fritz (2015)	Lab study: 3 change tasks (i.e., adding a new feature in JHotDraw) Field study: real-world setting; various tasks like elimination of performance bottleneck; implementing a user interface; implementation of test case	Java	–
Lee et al. (2016)	36 code comprehension tasks (partially taken from Siegmund et al., 2014)	Java	Small programs
Castelhano et al. (2018)	3 neutral programs, 3 programs with bugs	C	20–60
Crk and Kluthe (2016)	3 code comprehension tasks	Java	Small programs
Yamamoto et al. (2016)	12 tasks that required to find implementation strategy	Java	10–30 lines
Duraisingam et al. (2017)	12 code comprehension tasks	Java	–
Lee et al. (2017)	23 comprehension tasks (derived from Siegmund et al., 2014)	Java	Max. 20 lines
Kosti et al. (2018)	Pairs of 20 basic algorithms	C	–
Züger et al. (2018)	No specific task	–	–
Ahonen et al. (2016)	2 programming assignments	Java	–
Ahonen et al. (2018)	4–6 exercises with several sub-tasks	Java	–
Behroozi et al. (2018)	2 coding problems (on paper/whiteboard)	Could be chosen	–
Behroozi and Parnin (2018)	2 coding problems (on paper/whiteboard)	Could be chosen	–
Ahrens et al. (2019)	2 change tasks, 2 bug fixing tasks, and 2 extension tasks	Java	907 lines of code with 176 lines of comments
Couceiro et al. (2019a)	3 small programs with varying levels of difficulty	Java	13–49 lines
Couceiro et al. (2019b)	3 small programs with varying levels of difficulty	Java	13–49 lines
Couceiro et al. (2019c)	3 small programs with varying levels of difficulty	Java	13–49 lines
Doukakis (2019), Doukakis et al. (2020)	2 programming tasks	Python and Scratch	–
Fucci et al. (2019)	Comprehension task and Prose comprehension task (Text stimulus)	Java	20 lines
Huang et al. (2019)	Data structure manipulation (sequence, tree) and mental rotation	–	–
Ikramov et al. (2019)	Not specified	Not specified	Not specified
Ishida and Uwano (2019b)	16 tasks (8 easy and 8 difficult) consisting of understanding step and judgment step	Java	–
Ishida and Uwano (2019a)	16 comprehension tasks (8 easy and 8 difficult)	Java	–
Medeiros et al. (2019)	3 comprehension tasks with varying levels of difficulty	Java	–

(*continued on next page*)

Table 19 (*continued*).

ID	Task characteristics	Programming language	Size
Wulff-Jensen et al. (2019)	4 types of comprehension tasks (structural elements, textual elements, no such elements, both structural and textual)	Processing (builds upon Java)	11–28 lines
Girardi et al. (2020)	1 change tasks (Java program interacting with Stack Exchange API)	Java	–
Ikutani et al. (2020)	72 comprehension tasks from an open codeset provided by AIZU ONLINE JUDGE (http://judge.u-aizu.ac.jp/onlinejudge/)	Java	Average 26.4 +−2.4 lines and 59.3 +−17.1 characters per line
Ivanova et al. (2020)	Comprehension task (with 72 problems), spatial working memory localizer task, and a language localizer task Experiment 1 (Python): Python programs with English identifiers, Python programs with Japanese identifiers, sentence versions of programs 50% of the programs required math operations, 50% string operations; $\frac{1}{3}$ required sequential statements, $\frac{1}{2}$ if statements and $\frac{1}{3}$ loops Experiment 2 (Scratch): 2 conditions: short programs in Scratch, sentence version of these programs; 3 levels of difficulty	Python and Scratch	–
Liu et al. (2020)	Code comprehension, language/math/logic localizer, Multi-source interference task	Python	5 lines
Vrzakova et al. (2020)	Participants could choose pending code review task in CodeFlow	Depending on chosen task	Real code review tasks

References

Abid, N.J., Sharif, B., Dragan, N., Alrasheed, H., Maletic, J.I., 2019. Developer reading behavior while summarizing java methods: Size and context matters. In: 2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE). pp. 384–395. http://dx.doi.org/10.1109/ICSE.2019.00052.

Adam, M.T.P., Gimpel, H., Maedche, A., Riedl, R., 2017. Design blueprint for stress-sensitive adaptive enterprise systems. Bus. Inf. Syst. Eng. 59, 277–291. http://dx.doi.org/10.1007/s12599-016-0451-3.

Aghajani, H., Garbey, M., Omurtag, A., 2017. Measuring mental workload with EEG+fNIRS. Front. Hum. Neurosci. 11, 359. http://dx.doi.org/10.3389/fnhum.2017.00359.

Ahonen, L., Cowley, B.U., Hellas, A., Puolamäki, K., 2018. Biosignals reflect pair-dynamics in collaborative work: EDA and ECG study of pair-programming in a classroom environment. Sci. Rep. 8, 3138. http://dx.doi.org/10.1038/s41598-018-21518-3.

Ahonen, L., Cowley, B., Torniainen, J., Ukkonen, A., Vihavainen, A., Puolamäki, K., Cognitive Collaboration, 2016. Found in cardiac physiology: Study in classroom environment. PLOS ONE 11, 1–16. http://dx.doi.org/10.1371/journal.pone.0159178.

Ahrens, M., Schneider, K., Busch, M., 2019. Attention in software maintenance: An eye tracking study. In: Proceedings of the 6th International Workshop on Eye Movements in Programming (EMIP). IEEE, pp. 2–9. http://dx.doi.org/10.1109/EMIP.2019.00009.

Aschwanden, C., Crosby, M., 2006. Code scanning patterns in program comprehension. In: Proc. 39th Hawaii Int. Conf. Syst. Sci.

Badcock, N., Mousikou, B., Mahajan, Y., de Lissa, P., Thie, J., McArthur, G., 2013. Validation of the Emotiv EPOC® EEG gaming system for measuring research quality auditory ERPs. PeerJ 1, e38. http://dx.doi.org/10.7717/peerj.38.

Bastos, A.M., Schoffelen, J.-M., 2016. A tutorial review of functional connectivity analysis methods and their interpretational pitfalls. Front. Syst. Neurosci. 9, 175. http://dx.doi.org/10.3389/fnsys.2015.00175.

Baumgartner, D., Fischer, T., Riedl, R., Dreiseitl, S., 2019. Analysis of Heart Rate Variability (HRV) Feature Robustness for Measuring Technostress (NeuroIS Retreat 2018). In: Lecture Notes in Information Systems and Organisation, pp. 221–228. http://dx.doi.org/10.1007/978-3-030-01087-4_27.

Begel, A., 2016. Invited talk: Fun with software developers and biometrics. In: 2016 IEEE/ACM 1st International Workshop on Emotional Awareness in Software Engineering (SEmotion). pp. 1–2. http://dx.doi.org/10.1109/SEmotion.2016.009.

Behroozi, M., Lui, A., Moore, I., Ford, D., Parnin, C., 2018. Dazed: Measuring the cognitive load of solving technical interview problems at the whiteboard. In: Zisman, A., Apel, S. (Eds.), Proceedings of the 40th International Conference on Software Engineering New Ideas and Emerging Results - ICSE-NIER '18. ACM Press, New York, New York, USA, pp. 93–96. http://dx.doi.org/10.1145/3183399.3183415.

Behroozi, M., Parnin, C., 2018. Can we predict stressful technical interview settings through eye-tracking? In: Bednarik, R., Schulte, C. (Eds.), Proceedings of the Workshop on Eye Movements in Programming - EMIP '18. ACM Press, New York, New York, USA, pp. 1–5. http://dx.doi.org/10.1145/3216723.3216729.

Boucsein, W., Fowles, D., Grimnes, S., Ben-Shakhar, G., Roth, W., Dawson, M., Filion, D., 2012. Publication recommendations for electrodermal measurements. Psychophysiology 49, 1017–1034. http://dx.doi.org/10.1111/j.1469-8986.2012.01384.x.

Braithwaite, J.J., Watson, D.G., Jones, R., Rowe, M., 2015. A guide for analysing electrodermal activity (EDA) & skin conductance responses (SCRs) for psychological experiments (revised version 2.0).

Bronzino, J.D., 1995. The Biomedical Engineering HandBook. CRC Press, pp. 201–212.

Brown, J.A., Ivanov, V., Rogers, A., Succi, G., Tormasov, A., Yi, J., 2018. Toward a better understanding of how to develop software under stress - drafting the lines for future research. In: Damiani, E., Spanoudakis, G., Maciaszek, L.A. (Eds.), Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering, ENASE 2018, Funchal, Madeira, Portugal, March (2018). SciTePress, pp. 23–24. http://dx.doi.org/10.5220/0006794103980405.

Brown, J.A., Ivanov, V., Rogers, A., Succi, G., Tormasov, A., Yi, J., 2020. Toward a better understanding of how to develop software under stress - drafting the lines for future research.

Buckner, R., Krienen, F., Yeo, B.T.T., 2013. Opportunities and limitations of intrinsic functional connectivity MRI. Nat. Neurosci. 16, 832–837. http://dx.doi.org/10.1038/nn.3423.

Bunce, S., Izzetoglu, M., Izzetoglu, K., Onaral, B., Pourrezaei, K., 2006. Functional near-infrared spectroscopy. IEEE Eng. Med. Biol. Mag. Q. Mag. Eng. Med. Biol. Soc. 25, 54–62. http://dx.doi.org/10.1109/MEMB.2006.1657788.

Burattin, A., Kaiser, M., Neurauter, M., Weber, B., 2019. Learning process modeling phases from modeling interactions and eye tracking data. Data. Knowl. Eng. 121, 1–17. http://dx.doi.org/10.1016/j.datak.2019.04.001.

Busechian, S., Ivanov, V., Rogers, A., Sirazitdinov, I., Succi, G., Tormasov, A., Yi, J., 2018. Understanding the impact of pair programming on the minds of developers. In: Proceedings of the 40th International Conference on Software Engineering New Ideas and Emerging Results - ICSE-NIER '18. Presented at the 40th International Conference. ACM Press, Gothenburg, Sweden, pp. 85–88. http://dx.doi.org/10.1145/3183399.3183413.

Buxi, D., Redoute, J.-M., Yuce, M., 2015. A survey on signals and systems in ambulatory blood pressure monitoring using pulse transit time. Physiol. Meas. 36, R1–R26. http://dx.doi.org/10.1088/0967-3334/36/3/R1.

Castelhano, J., Duarte, I.C., Ferreira, C., Duraes, J., Madeira, H., Castelo-Branco, M., 2018. The role of the insula in intuitive expert bug detection in computer code: an fMRI study. Brain Imaging Behav. http://dx.doi.org/10.1007/s11682-018-9885-1.

Couceiro, R., Carvalho, P., Castelo Branco, M., Madeira, H., Barbosa, R., Duraes, J., Duarte, G., Castelhano, J., Duarte, C., Teixeira, C., Laranjeiro, N., Medeiros, J., 2019a. Spotting problematic code lines using nonintrusive programmers' biofeedback. In: Proceedings of the 30th International Symposium on Software Reliability Engineering (ISSRE). IEEE, pp. 93–103. http://dx.doi.org/10.1109/ISSRE.2019.00019.

Couceiro, R., Duarte, G., Duraes, J., Castelhano, J., Duarte, C., Teixeira, C., Castelo Branco, M., Carvalho, P., Madeira, H., 2019b. Pupillography as indicator of programmers' mental effort and cognitive overload. In: Proceedings of the 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN). IEEE, pp. 638–644. http://dx.doi.org/10.1109/DSN.2019.00069.

Couceiro, R., Duarte, G., Duraes, J., Castelhano, J., Duarte, C., Teixeira, C., Castelo Branco, M., Carvalho, P., Madeira, H., 2019c. Biofeedback augmented software engineering: Monitoring of programmers' mental effort. In: Proceedings of the 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER). IEEE, pp. 37–40. http://dx.doi.org/10.1109/ICSE-NIER.2019.00018.

Cowley, B., Filetti, M., Lukander, K., Torniainen, J., Henelius, A., Ahonen, L., Barral, O., Kosunen, I., Valtonen, T., Huotilainen, M., Ravaja, N., Jacucci, G., 2016. The psychophysiology primer: A guide to methods and a broad review with a focus on human–computer interaction. Found. Trends® Hum.-Comput. Interact. 9, 151–308. http://dx.doi.org/10.1561/1100000065.

Crk, I., Kluthe, T., 2014. Toward using alpha and theta brain waves to quantify programmer expertise. In: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Presented at the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, Chicago, IL, pp. 5373–5376. http://dx.doi.org/10.1109/EMBC.2014.6944840.

Crk, I., Kluthe, T., 2016. Assessing the contribution of the individual alpha frequency (IAF) in an EEG-based study of program comprehension. In: Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS 2016-Octob. pp. 4601–4604. http://dx.doi.org/10.1109/EMBC.2016.7591752.

Crk, I., Kluthe, T., Stefik, A., 2016. Understanding programming expertise: An empirical study of phasic brain wave changes. ACM Trans. Comput.-Hum. Interact. 23. http://dx.doi.org/10.1145/2829945.

Dawson, M.E., Schell, A.M., Filion, D.L., 2016. The electrodermal system. In: Cacioppo, J.T., Tassinary, L.G., Berntson, G.G.E. (Eds.), In: Handbook of Psychophysiology, Cambridge Handbooks in Psychology, Cambridge University Press, pp. 217–243. http://dx.doi.org/10.1017/9781107415782.010.

Dimoka, A., Banker, R.D., Benbasat, I., Davis, F.D., Dennis, A.R., Gefen, D., Gupta, A., Ischebeck, A., Kenning, P., Müller-Putz, G.R., Pavlou, P.A., Riedl, R., vom Brocke, J., Weber, B., 2012. On the use of neurophysiological tools in IS research: Developing a research agenda for NeuroIS. MIS Q 36 (3), 679–702.

Djamasbi, S., 2014. Eye tracking and web experience. AIS Trans. Hum.-Comput. Interact. 6, 37–54. http://dx.doi.org/10.17705/1thci.00060.

Doukakis, S., 2019. Exploring brain activity and transforming knowledge in visual and textual programming using neuroeducation approaches. AIMS Neurosci. 6, 175–190. http://dx.doi.org/10.3934/Neuroscience.2019.3.175.

Doukakis, S., Papalaskari, M.-A., Vlamos, P., Plerou, A., Giannopoulou, P., 2020. Undergraduate students' brain activity in visual and textual programming. In: Vlamos, P. (Ed.), GeNeDis 2018. Advances in Experimental Medicine and Biology, vol. 1194. Springer, Cham, http://dx.doi.org/10.1007/978-3-030-32622-7_40.

Duchowski, A., 2017. Eye Tracking Methodology: Theory and Practice, third ed. Springer, http://dx.doi.org/10.1007/978-3-319-57883-5.

Duchowski, A.T., Krejtz, K., Gehrer, N.A., Bafna, T., Bækgaard, P., 2020. The low/high index of pupillary activity. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI '20. Association for Computing Machinery, New York, NY, USA, pp. 1–12. http://dx.doi.org/10.1145/3313831.3376394.

Duchowski, A.T., Krejtz, K., Krejtz, I., Biele, C., Niedzielska, A., Kiefer, P., Raubal, M., Giannopoulos, I., 2018. The index of pupillary activity: Measuring cognitive load Vis-à-vis task difficulty with pupil oscillation. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18. ACM, New York, NY, USA, pp. 282:1–282:13. http://dx.doi.org/10.1145/3173574.3173856.

Duraes, J., Madeira, H., Castelhano, J., Duarte, C., Branco, M.C., 2016. WAP: Understanding the brain at software debugging. In: 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE). Presented at the 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE). IEEE, Ottawa, ON, Canada, pp. 87–92. http://dx.doi.org/10.1109/ISSRE.2016.53.

Duraisingam, A., Palaniappan, R., Andrews, S., 2017. Cognitive task difficulty analysis using EEG and data mining. In: 2017 Conference on Emerging Devices and Smart Systems (ICEDSS). Presented at the 2017 Conference on Emerging Devices and Smart Systems (ICEDSS). IEEE, Mallasamudram, Tiruchengode, India, pp. 52–57. http://dx.doi.org/10.1109/ICEDSS.2017.8073658.

Duvinage, M., Castermans, T., Dutoit, T., Petieau, M., Hoellinger, T., Saedeleer, C., Seetharaman, K., Cheron, G., 2012. A P300-based quantitative comparison between the emotiv epoc headset and a medical EEG device. In: Biomedical Engineering Online. http://dx.doi.org/10.2316/P.2012.764-071.

Elgendi, M., 2012. On the analysis of fingertip photoplethysmogram signals. Curr. Cardiol. Rev. 8, 14–25. http://dx.doi.org/10.2174/157340312801215782.

Fakhoury, S., 2018. Moving towards objective measures of program comprehension. In: Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering - ESEC/FSE 2018. Presented at the 2018 26th ACM Joint Meeting. ACM Press, Lake Buena Vista, FL, USA, pp. 936–939. http://dx.doi.org/10.1145/3236024.3275426.

Fakhoury, S., Ma, Y., Arnaoudova, V., Adesope, O., 2018. The effect of poor source code lexicon and readability on developers' cognitive load. In: Proceedings of the 26th Conference on Program Comprehension - ICPC '18. Presented at the 26th Conference. ACM Press, Gothenburg, Sweden, pp. 286–296. http://dx.doi.org/10.1145/3196321.3196347.

Fakhoury, S., Roy, D., Ma, Y., Arnaoudova, V., Adesope, O., 2020. Measuring the impact of lexical and structural inconsistencies on developers' cognitive load during bug localization. Empir. Softw. Eng. 25, 2140–2178. http://dx.doi.org/10.1007/s10664-019-09751-4.

Fischer, T., Halmerbauer, G., Meyr, E., Riedl, R., 2017. Blood pressure measurement: A classic of stress measurement and its role in technostress research. In: Lecture Notes in Information Systems and Organisation, http://dx.doi.org/10.1007/978-3-319-67431-5_4.

Floyd, B., Santander, T., Weimer, W., 2017. Decoding the representation of code in the brain: An fMRI study of code review and expertise. In: 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE). Presented at the 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE). IEEE, Buenos Aires, pp. 175–186. http://dx.doi.org/10.1109/ICSE.2017.24.

Ford, D., Barik, T., Parnin, C., 2016. Studying sustained attention and cognitive states with eye tracking in remote technical interviews. In: Proceedings of the Third International Workshop on Eye Movements in Programming.

Fountaine, A., Sharif, B., 2017. Emotional awareness in software development: Theory and measurement. In: 2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion). Presented at the 2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion). IEEE, Buenos Aires, Argentina, pp. 28–31. http://dx.doi.org/10.1109/SEmotion.2017.12.

Friston, K.J., 1994. Functional and effective connectivity in neuroimaging: A synthesis. Hum. Brain Mapp. 2, 56–78. http://dx.doi.org/10.1002/hbm.460020107.

Friston, K., 2002. Functional integration and inference in the brain. Prog. Neurobiol. 68, 113–143. http://dx.doi.org/10.1016/S0301-0082(02)00076-X.

Fritz, T., Begel, A., Müller, S.C., Yigit-Elliott, S., Züger, M., 2014. Using psychophysiological measures to assess task difficulty in software development. In: Proceedings of the 36th International Conference on Software Engineering - ICSE 2014. Presented at the 36th International Conference. ACM Press, Hyderabad, India, pp. 402–413. http://dx.doi.org/10.1145/2568225.2568266.

Fritz, T., Muller, S.C., 2016. Leveraging biometric data to boost software developer productivity. In: 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER). Presented at the 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER). IEEE, Suita, Osaka, Japan, pp. 66–77. http://dx.doi.org/10.1109/SANER.2016.107.

Fucci, D., Girardi, D., Novielli, N., Quaranta, L., Lanubile, F., 2019. A replication study on code comprehension and expertise using lightweight biometric sensors. In: Proceedings of the 27th International Conference on Program Comprehension (ICPC). IEEE, pp. 311–322. http://dx.doi.org/10.1109/ICPC.2019.00050.

Gefen, D., Ayaz, H., Onaral, B., 2014. Applying functional near infrared (fNIR) spectroscopy to enhance MIS research. AIS Trans. Hum.-Comput. Interact. 6, 55–73. http://dx.doi.org/10.17705/1thci.00061.

Gefen, D., Riedl, R., 2018. Adding background music as new stimuli of interest to information systems research. Eur. J. Inf. Syst. 27, 46–61. http://dx.doi.org/10.1080/0960085X.2017.1387713.

Girardi, D., Lanubile, F., Novielli, N., Fucci, D., 2018. Sensing developers' emotions: the design of a replicated experiment. In: Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering - SEmotion '18. Presented at the 3rd International Workshop. ACM Press, Gothenburg, Sweden, pp. 51–54. http://dx.doi.org/10.1145/3194932.3194940.

Girardi, D., Lanubile, F., Novielli, N., Quaranta, L., Serebrenik, A., 2019. Towards recognizing the emotions of developers using biometrics: The design of a field study. In: Proceedings of the 4th International Workshop on Emotion Awareness in Software Engineering (SEmotion). IEEE, pp. 13–16. http://dx.doi.org/10.1109/SEmotion.2019.00010.

Girardi, D., Novielli, N., Fucci, D., Lanubile, F., 2020. Recognizing developers' emotions while programming. In: Presented at the ICSE, Vol. 14. pp. 666–677. http://dx.doi.org/10.1145/3377811.3380374.

Goldberg, J., 2012. Relating perceived web page complexity to emotional valence and eye movement metrics. Proc. Hum. Factors Ergon. Soc. Annu. Meet 56, 501–505. http://dx.doi.org/10.1177/1071181312561104.

Goldberg, J.H., 2014. Measuring software screen complexity: Relating eye tracking, emotional valence, and subjective ratings. Int. J. Hum. Comput. Interact. 30, 518–532. http://dx.doi.org/10.1080/10447318.2014.906156.

Goncales, L., Farias, K., da Silva, B., Fessler, J., 2019. Measuring the cognitive load of software developers: A systematic mapping study. In: Proceedings of the 27th International Conference on Program Comprehension (ICPC). IEEE, pp. 42–52. http://dx.doi.org/10.1109/ICPC.2019.00018.

González, V.M., Robbes, R., Góngora, G., Medina, S., 2015. A034 - measuring concentration while programming with low-cost BCI devices: Differences between debugging and creativity tasks. In: Schmorrow, D.D., Fidopiastis, C.M. (Eds.), Foundations of Augmented Cognition. Springer International Publishing, Cham, pp. 605–615. http://dx.doi.org/10.1007/978-3-319-20816-9_58.

Guarnera, D.T., Bryant, C.A., Mishra, A., Maletic, J.I., Sharif, B., 2018. iTrace: eye tracking infrastructure for development environments. In: Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, ETRA 2018, Warsaw, Poland, June 14-17. (2018), pp. 105:1–105:3. http://dx.doi.org/10.1145/3204493.3208343.

Harmon-Jones, E., Beer, J.S., 2009. Methods in Social Neuroscience. Guilford Publications.

Helmholtz, H., 1853. Ueber einige Gesetze der Vertheilung elektrischer Ströme in körperlichen Leitern mit Anwendung auf die thierisch-elektrischen Versuche. Ann. Phys. 165, 211–233. http://dx.doi.org/10.1002/andp.18531650603.

Holmqvist, K., Andersson, R., 2017. Eye-tracking: A comprehensive guide to methods, paradigms and measures.

Huang, Y., Liu, X., Krueger, R., Santander, T., Hu, X., Leach, K., Weimer, W., 2019. Distilling neural representations of data structure manipulation using fMRI and fNIRS. In: Proceedings of the 41st International Conference on Software Engineering (ICSE). IEEE, pp. 396–407. http://dx.doi.org/10.1109/ICSE.2019.00053.

Huang, S., Miranda, P., 2015. Incorporating human intention into self-adaptive systems. In: Proceedings of the 37th International Conference on Software Engineering - Volume 2, ICSE '15. IEEE Press, Piscataway, NJ, USA, pp. 571–574.

Huang, S., Tognoli, E., 2014. Brainware: Synergizing software systems and neural inputs. In: Companion Proceedings of the 36th International Conference on Software Engineering, ICSE Companion 2014. ACM, New York, NY, USA, pp. 444–447. http://dx.doi.org/10.1145/2591062.2591131.

Hubert, Marco, Linzmajer, M., Riedl, R., Hubert, Mirja, Kenning, P., Weber, B., 2017. Using psycho-physiological interaction analysis with fMRI data in IS research: A guideline. Commun. Assoc. Inf. Syst. 40, 181–217. http://dx.doi.org/10.17705/1CAIS.04009.

Huyer, S., 2015. UNESCO Science Report: Towards 2030. Is the gender gap narrowing in science and engineering?

Işbilir, E., Çakır, M.P., Acartürk, C., Tekerek, A.Ş., 2019. Towards a multimodal model of cognitive workload through synchronous optical brain imaging and eye tracking measures. Front. Hum. Neurosci. 13, 375. http://dx.doi.org/10.3389/fnhum.2019.00375.

Ikramov, R., Ivanov, V., Masyagin, S., Shakirov, R., Sirazidtinov, I., Succi, G., Thapaliya, A., Tormasov, A., Zufarova, O., 2019. Initial evaluation of the brain activity under different software development situations. In: Proceedings of the 31st International Conference on Software Engineering and Knowledge Engineering, International Conferences on Software Engineering and Knowledge Engineering. KSI Research Inc. and Knowledge Systems Institute Graduate School, pp. 741–747. http://dx.doi.org/10.18293/SEKE2019-215.

Ikutani, Y., Kubo, T., Nishida, S., Hata, H., Matsumoto, K., Ikeda, K., Nishimoto, S., 2020. Expert programmers have fine-tuned cortical representations of source code. eNeuro. http://dx.doi.org/10.1523/ENEURO.0405-20.2020.

Ikutani, Y., Uwano, H., 2014. Brain activity measurement during program comprehension with NIRS. Int. J. Netw. Distrib. Comput. 2, 259–268. http://dx.doi.org/10.2991/ijndc.2014.2.4.7.

Ishida, T., Uwano, H., 2019a. Synchronized analysis of eye movement and EEG during program comprehension. In: Proceedings of the 6th International Workshop on Eye Movements in Programming (EMIP). IEEE, pp. 26–32. http://dx.doi.org/10.1109/EMIP.2019.00012.

Ishida, T., Uwano, H., 2019b. Time series analysis of programmer's EEG for debug state classification. In: Marr, S., Cazzola, W. (Eds.), Proceedings of the 3rd International Companion Conference on Art, Science, and Engineering of Programming - Programming '19. ACM Press, New York, New York, USA, pp. 1–7. http://dx.doi.org/10.1145/3328433.3328459.

Ivanova, A.A., Srikant, S., Sueoka, Y., Kean, H.H., Dhamala, R., O'Reilly, U.-M., Bers, M.U., Fedorenko, E., 2020. Comprehension of computer code relies primarily on domain-general executive brain regions. eLife 9, e58906. http://dx.doi.org/10.7554/eLife.58906.

Jbara, A., Feitelson, D.G., 2015. How programmers read regular code: A controlled experiment using eye tracking. In: 2015 IEEE 23rd International Conference on Program Comprehension. Presented at the 2015 IEEE 23rd International Conference on Program Comprehension (ICPC). IEEE, Florence, Italy, pp. 244–254. http://dx.doi.org/10.1109/ICPC.2015.35.

Jennings, J.R., Allen, B., 2016. In: Cacioppo, J.T., Tassinary, L.G., Berntson, G.G.E. (Eds.), Handbook of Psychophysiology, Cambridge University Press, Cambridge.

Jimenez-Molina, A., Retamal, C., Lira, H., 2018. Using psycho-physiological sensors to assess mental workload in web browsing. Sensors 18. http://dx.doi.org/10.3390/s18020458.

Just, M., Carpenter, P., 1980. A theory of reading: from eye fixations to comprehension. Psychol. Rev. 87, 329–354.

Kandel, E.R., Schwartz, J.H., Jessell, T.M., Hudspeth, S.A.S.A.J., 2013. Principles of Neural Science, fifth ed. McGraw-Hill.

Kanoga, S., Nakanishi, M., Mitsukura, Y., 2016. Assessing the effects of voluntary and involuntary eyeblinks in independent components of electroencephalogram. Neurocomputing 193, 20–32. http://dx.doi.org/10.1016/j.neucom.2016.01.057.

Kim, E., Kim, D.-S., Ahmad, F., Park, H., 2013. Pattern-based granger causality mapping in fMRI. Brain Connect. 3, http://dx.doi.org/10.1089/brain.2013.0148.

Kitchenham, B., Charters, S., 2007. Guidelines for performing systematic literature reviews in software engineering.

Kontogiorgos, D., Manikas, K., 2015. Towards identifying programming expertise with the use of physiological measures. In: Proc. of the Int'l Workshop on Eye Movements in Programming.

Kosti, M.V., Georgiadis, K., Adamos, D.A., Laskaris, N., Spinellis, D., Angelis, L., 2018. Towards an affordable brain computer interface for the assessment of programmers' mental workload. Int. J. Hum.-Comput. Stud. 115, 52–66. http://dx.doi.org/10.1016/j.ijhcs.2018.03.002.

Kwong, K.K., Belliveau, J.W., Chesler, D.A., Goldberg, I., Weisskoff, R.M., Poncelet, B.P., Kennedy, D., Hoppel, B., Cohen, M., Turner, R., 1992. Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation. Proc. Natl. Acad. Sci. USA 89, 5675–5679. http://dx.doi.org/10.1073/pnas.89.12.5675.

Lang, P., Bradley, M., Cuthbert, B., 1990. Emotion, attention, and the startle reflex. Psychol. Rev. 97, 377—395. http://dx.doi.org/10.1037/0033-295x.97.3.377.

Lee, S., Hooshyar, D., Ji, H., Nam, K., Lim, H., 2017. Mining biometric data to predict programmer expertise and task difficulty. Clust. Comput. 1–11. http://dx.doi.org/10.1007/s10586-017-0746-2.

Lee, S., Matteson, A., Hooshyar, D., Kim, S., Jung, J., Nam, G., Lim, H., 2016. Comparing programming language comprehension between novice and expert programmers using EEG analysis. In: 2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE). Presented at the 2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE). IEEE, Taichung, Taiwan, pp. 350–355. http://dx.doi.org/10.1109/BIBE.2016.30.

Léger, P.-M., Senecal, S., Courtemanche, F., Guinea, A., Titah, R., Fredette, M., L. LeMoyne, E., 2014. Precision is in the eye of the beholder: Application of eye fixation-related potentials to information systems research. J. Assoc. Inf. Syst. 15. http://dx.doi.org/10.17705/1jais.00376.

Lekkala, R.G.R., Kuntamalla, S., 2017. A novel approach for comparison of heart rate variability derived from synchronously measured electrocardiogram and photoplethysmogram. In: 2017 IEEE Healthcare Innovations and Point of Care Technologies (HI-POCT). pp. 121–124. http://dx.doi.org/10.1109/HIC.2017.8227599.

Li, Y., Yu, D., 2018. Variations of the functional brain network efficiency in a Young clinical sample within the autism spectrum: A fNIRS investigation. Front. Physiol. 9, 67. http://dx.doi.org/10.3389/fphys.2018.00067.

Lin, T., Imamiya, A., 2006. Evaluating usability based on multimodal information: An empirical study. In: Proceedings of the 8th International Conference on Multimodal Interfaces, ICMI '06. ACM, New York, NY, USA, pp. 364–371. http://dx.doi.org/10.1145/1180995.1181063.

Liu, Y., Ayaz, H., Shewokis, P.A., 2017. Multisubject learning for mental workload classification using concurrent EEG, fNIRS, and physiological measures. Front. Hum. Neurosci. 11, 389. http://dx.doi.org/10.3389/fnhum.2017.00389.

Liu, Y.-F., Kim, J., Wilson, C., Bedny, M., 2020. Computer code comprehension shares neural resources with formal logical inference in the fronto-parietal network. eLife 9, e59340. http://dx.doi.org/10.7554/eLife.59340.

Logothetis, N., 2008. What we can and cannot do with fMRI. Nature 453, 869–878. http://dx.doi.org/10.1038/nature06976.

Logothetis, N., Pauls, J., Augath, M.A., Trinath, T., Oeltermann, A., 2001. Neurophysiological investigation of the basis of the fMRI signal. Nature 412, 150–157. http://dx.doi.org/10.1038/35084005.

Lystad, R.P., Pollard, H., 2009. Functional neuroimaging: a brief overview and feasibility for use in chiropractic research. J. Can. Chiropr. Assoc. 53, 59–72.

Mack, S., Kandel, E.R., Jessell, T.M., Schwartz, J.H., Siegelbaum, S.A., Hudspeth, A.J., 2013. Principles of Neural Science, fifth ed. Principles of Neural Science. McGraw-Hill Education.

McDuff, D., Nishidate, I., Nakano, K., Haneishi, H., Aoki, Y., Tanabe, C., Niizeki, K., Aizu, Y., 2020. Non-contact imaging of peripheral hemodynamics during cognitive and psychological stressors. Sci. Rep. 10. http://dx.doi.org/10.1038/s41598-020-67647-6.

Medeiros, J., Couceiro, R., Castelhano, J., Castelo Branco, M., Duarte, G., Duarte, C., Duraes, J., Madeira, H., Carvalho, P., Teixeira, C., 2019. Software code complexity assessment using EEG features. In: Proceedings of the 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 1413–1416. http://dx.doi.org/10.1109/EMBC43219.2019.

Menzen, J.P., Farias, K., Bischoff, V., 2020. Using biometric data in software engineering: A systematic mapping study. Behav. Inf. Technol. 17, 1–23. http://dx.doi.org/10.1080/0144929X.2020.1734086.

Mi, Q., Keung, J., Huang, J., Xiao, Y., 2017. Using eye tracking technology to analyze the impact of stylistic inconsistency on code readability. In: 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). Presented at the 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, Prague, Czech Republic, pp. 579–580. http://dx.doi.org/10.1109/QRS-C.2017.102.

Michel, C.M., Murray, M.M., Lantz, G., Gonzalez, S., Spinelli, L., de Peralta, R.G., 2004. EEG source imaging. Clin. Neurophysiol. 115, 2195–2222. http://dx.doi.org/10.1016/j.clinph.2004.06.001.

Minas, R., Kazman, R., Tempero, E., 2017. Neurophysiological impact of software design processes on software developers. In: Schmorrow, D.D., Fidopiastis, C.M. (Eds.), Augmented Cognition. Enhancing Cognition and Behavior in Complex Human Environments. Springer International Publishing, Cham, pp. 56–64.

Moffat, D.C., Paterson, J.H., 2016. Eye-tracking to trace anxieties of programmers. In: Proceedings of the 16th International Conference on Bioinformatics and Bioengineering.

Müller, S.C., 2015. Measuring software developers' perceived difficulty with biometric sensors. In: 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering. pp. 887–890. http://dx.doi.org/10.1109/ICSE.2015.284.

Müller, S.C., Fritz, T., 2015. 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 (2015) 16-24, Vol. 1. pp. 688–699. http://dx.doi.org/10.1109/ICSE.2015.334.

Müller, S.C., Fritz, T., 2016. Using (bio)metrics to predict code quality online. In: Proceedings of the 38th International Conference on Software Engineering - ICSE '16. Presented at the 38th International Conference. ACM Press, Austin, Texas, pp. 452–463. http://dx.doi.org/10.1145/2884781.2884803.

Müller-Putz, G., Riedl, R., Wriessnegger, S., 2015. Electroencephalography (EEG) as a research tool in the information systems discipline: Foundations, measurement, and applications. Commun. Assoc. Inf. Syst. 37, 911–948. http://dx.doi.org/10.17705/1CAIS.03746.

Nakagawa, T., Kamei, Y., Uwano, H., Monden, A., Matsumoto, K., German, D.M., 2014. Quantifying programmers' mental workload during program comprehension based on cerebral blood flow measurement: a controlled experiment. In: Companion Proceedings of the 36th International Conference on Software Engineering - ICSE Companion 2014. Presented at the Companion the 36th International Conference. ACM Press, Hyderabad, India, pp. 448–451. http://dx.doi.org/10.1145/2591062.2591098.

Nakano, T., Kato, M., Morito, Y., Itoi, S., Kitazawa, S., 2012. Blink-related momentary activation of the default mode network while viewing videos. Proc. Natl. Acad. Sci. USA 110. http://dx.doi.org/10.1073/pnas.1214804110.

Naqvi, N., Bechara, A., 2006. Skin conductance: a psychophysiological approach to the study of decision making. Methods Mind. 103–122.

Newman, A., 2019. Research Methods for Cognitive Neuroscience. SAGE Publications.

Nolan, K., Mooney, A., Bergin, S., 2016. Examining the role of cognitive load when learning to program. In: Proceedings of the 16th International Conference on Bioinformatics and Bioengineering.

Novielli, N., Begel, A., Maalej, W., 2018. Introduction to the special issue on affect awareness in software engineering. J. Syst. Softw. 148. http://dx.doi.org/10.1016/j.jss.2018.11.016.

Novielli, N., Serebrenik, A., 2019. Sentiment and emotion in software engineering. IEEE Softw. 36, 6–23. http://dx.doi.org/10.1109/MS.2019.2924013.

Obaidellah, U., Al Haek, M., Cheng, P.C.-H., 2018. A survey on the usage of eye-tracking in computer programming. ACM Comput. Surv. 51. http://dx.doi.org/10.1145/3145904.

O'Reilly, J., Woolrich, M., Behrens, T., Smith, S., Johansen-Berg, H., 2012. Tools of the trade: Psychophysiological interactions and functional connectivity. Soc. Cogn. Affect. Neurosci. 7, 604–609. http://dx.doi.org/10.1093/scan/nss055.

Ostberg, J.-P., Graziotin, D., Wagner, S., Derntl, B., 2017. Towards the assessment of stress and emotional responses of a salutogenesis-enhanced software tool using psychophysiological measurements. In: 2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion). Presented at the 2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion). IEEE, Buenos Aires, Argentina, pp. 22–25. http://dx.doi.org/10.1109/SEmotion.2017.4.

Parnin, C., 2010. A cognitive neuroscience perspective on memory for programming tasks. In: the Proceedings of the 22nd Annual Meeting of the Psychology of Programming Interest Group (PPIG).

Parnin, C., 2011. Subvocalization - toward hearing the inner thoughts of developers. In: 2011 IEEE 19th International Conference on Program Comprehension. Presented at the 2011 IEEE 19th International Conference on Program Comprehension (ICPC). IEEE, Kingston, ON, Canada, pp. 197–200. http://dx.doi.org/10.1109/ICPC.2011.49.

Peitek, N., 2018. A neuro-cognitive perspective of program comprehension. In: Proceedings of the 40th International Conference on Software Engineering Companion Proceeedings - ICSE '18. Presented at the 40th International Conference. ACM Press, Gothenburg, Sweden, pp. 496–499. http://dx.doi.org/10.1145/3183440.3183442.

Peitek, N., Apel, S., Brechmann, A., Parnin, C., Siegmund, J., 2019. CodersMUSE: Multi-modal data exploration of program-comprehension experiments. In: Proceedings of the 27th International Conference on Program Comprehension (ICPC). IEEE, pp. 126–129. http://dx.doi.org/10.1109/ICPC.2019.00027.

Peitek, N., Siegmund, J., Apel, S., Kastner, C., Parnin, C., Bethmann, A., Leich, T., Saake, G., Brechmann, A., 2018a. A look into programmers' heads. IEEE Trans. Softw. Eng. 5589, 1–20. http://dx.doi.org/10.1109/TSE.2018.2863303.

Peitek, N., Siegmund, J., Brechmann, A., 2017. Enhancing fMRI studies of program comprehension with eye-tracking. In: Proc. Int'l Workshop on Eye Movements in Programming.

Peitek, N., Siegmund, J., Parnin, C., Apel, S., Brechmann, A., 2018b. Beyond gaze: preliminary analysis of pupil dilation and blink rates in an fMRI study of program comprehension. In: Proceedings of the Workshop on Eye Movements in Programming - EMIP '18. Presented at the Workshop. ACM Press, Warsaw, Poland, pp. 1–5. http://dx.doi.org/10.1145/3216723.3216726.

Peitek, N., Siegmund, J., Parnin, C., Apel, S., Brechmann, A., 2018c. Toward conjoint analysis of simultaneous eye-tracking and fMRI data for program-comprehension studies. In: Proceedings of the Workshop on Eye Movements in Programming - EMIP '18. Presented at the Workshop. ACM Press, Warsaw, Poland, pp. 1–5. http://dx.doi.org/10.1145/3216723.3216725.

Peitek, N., Siegmund, J., Parnin, C., Apel, S., Hofmeister, J.C., Brechmann, A., 2018d. Simultaneous measurement of program comprehension with fMRI and eye tracking: a case study. In: Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '18. Presented at the 12th ACM/IEEE International Symposium. ACM Press, Oulu, Finland, pp. 1–10. http://dx.doi.org/10.1145/3239235.3240495.

Peitek, N., Siegmund, J., Parnin, C., Apel, S., Hofmeister, J.C., Kästner, C., Begel, A., Bethmann, A., Brechmann, A., 2018e. Neural efficiency of top-down program comprehension. In: Tichy, M., Bodden, E., Kuhrmann, M., Wagner, S., Steghöfer, J.-P. (Eds.), Software Engineering and Software Management 2018. In: GI-Edition: Lecture Notes in Informatics. Proceedings, Gesellschaft für Informatik e.V. (GI), Bonn, pp. 151–152.

Penny, W.D., Stephan, K.E., Mechelli, A., Friston, K.J., 2004. Modelling functional integration: a comparison of structural equation and dynamic causal models. NeuroImage 23, S264–S274. http://dx.doi.org/10.1016/j.neuroimage.2004.07.041.

Phukan, A., 2009. Measuring usability via biometrics. In: Proceedings of the 3D International Conference on Online Communities and Social Computing: Held As Part of HCI International 2009, OCSC '09. Springer-Verlag, Berlin, Heidelberg, pp. 101–107. http://dx.doi.org/10.1007/978-3-642-02774-1_11.

Posada-Quintero, H., Chon, K., 2020. Innovations in electrodermal activity data collection and signal processing: A systematic review. Sensors 20, 479. http://dx.doi.org/10.3390/s20020479.

Quaresima, V., Ferrari, M., 2016. Functional near-infrared spectroscopy (fNIRS) for assessing cerebral cortex function during human behavior in natural/social situations: A concise review. Organ. Res. Methods 22. http://dx.doi.org/10.1177/1094428116658959.

Radevski, S., Hata, H., Matsumoto, K., 2015. Real-time monitoring of neural state in assessing and improving software developers' productivity. pp. 93–96. http://dx.doi.org/10.1109/CHASE.2015.28.

Raichle, M.E., MacLeod, A.M., Snyder, A.Z., Powers, W.J., Gusnard, D.A., Shulman, G.L., 2001. A default mode of brain function. Proc. Natl. Acad. Sci. 98, 676–682. http://dx.doi.org/10.1073/pnas.98.2.676.

Rajala, S., Lindholm, H., Taipalus, T., 2018. Comparison of photoplethysmogram measured from wrist and finger and the effect of measurement location on pulse arrival time. Physiol. Meas. 39. http://dx.doi.org/10.1088/1361-6579/aac7ac.

Riedl, R., Banker, R., Benbasat, I., Davis, F., Dennis, A., Dimoka, A., Gefen, D., Gupta, A., Ischebeck, A., Kenning, P., et al., 2010a. On the foundations of NeuroIS: reflections on the gmunden retreat 2009. Commun. Assoc. Inf. Syst. 27, 15.

Riedl, R., Davis, F.D., Banker, R.D., Kenning, P.H., 2017a. Neuroscience in information systems research: Applying knowledge of brain fuems and organisation: Applying knowledge of brain functionality without neuroscience tools. In: Lecture Notes in Information Systems, Springer, Berlin.

Riedl, R., Davis, F., Hevner, A., 2014. Towards a NeuroIS research methodology: Intensifying the discussion on methods, tools, and measurement. J. Assoc. Inf. Syst. 15, 1–25. http://dx.doi.org/10.17705/1jais.00377.

Riedl, R., Fischer, T., Léger, P.-M., 2017b. A decade of neurois research: status quo, challenges, and future directions. In: Proceedings of the International Conference on Information Systems - Transforming Society with Digital Innovation, {ICIS} 2017, Seoul, South Korea, December 2017. pp. 10–13.

Riedl, R., Fischer, T., Léger, P.-M., Davis, F., 2020a. A decade of neurois research: Progress, challenges, and future directions. DATA BASE Adv. Inf. Syst. 51, 13–54.

Riedl, R., Hubert, M., Kenning, P., 2010b. Are there neural gender differences in online trust? An fMRI study on the perceived trustworthiness of ebay offers. MIS Q 34, 397–428.

Riedl, R., Léger, P.-M., 2016. Fundamentals of NeuroIS - Information Systems and the Brain, Studies in Neuroscience, Psychology and Behavioral Economics. Springer, http://dx.doi.org/10.1007/978-3-662-45091-8.

Riedl, R., Minas, R.K., Dennis, A.R., Müller-Putz, G.R., 2020b. Consumer-grade EEG instruments: Insights on the measurement quality based on a literature review and implications for neurois research. In: Davis, F.D., Riedl, R., vom Brocke, J., Léger, P.-M., Randolph, A.B., Fischer, T. (Eds.), Information Systems and Neuroscience - NeuroIS Retreat 2020, Lecture Notes in Information Systems and Organisation. Springer Science and Business Media Deutschland GmbH, Germany, pp. 350–361. http://dx.doi.org/10.1007/978-3-030-60073-0_41.

Riseberg, J., Klein, J., Fernandez, R., Picard, R.W., 1998. Frustrating the user on purpose: Using biosignals in a pilot study to detect the user's emotional state. In: CHI 98 Conference Summary on Human Factors in Computing Systems, CHI '98. Association for Computing Machinery, New York, NY, USA, pp. 227–228. http://dx.doi.org/10.1145/286498.286715.

Rogers, B., Morgan, V., Newton, A., Gore, J., 2008. Assessing functional connectivity in the human brain by fMRI. Magn. Reson. Imaging 25, 1347–1357. http://dx.doi.org/10.1016/j.mri.2007.03.007.

Roy, D., Fakhoury, S., Arnaoudova, V., 2020. VITALSE: visualizing eye tracking and biometric data. In: Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings.

Rutland, J., Brown, S., Feldman, R., Sharma, H., Markowitz, M., Schneider, M., Delman, B., Murrough, J., Balchandani, P., 2019. Hippocampal subfield-specific connectivity findings in major depressive disorder: A 7 T diffusion MRI study. J. Psychiatr. Res. 111. http://dx.doi.org/10.1016/j.jpsychires.2019.02.008.

Sánchez-Gordón, M., Colomo-Palacios, R., 2019. Taking the emotional pulse of software engineering — A systematic literature review of empirical studies. Inf. Softw. Technol. 115, 23–43. http://dx.doi.org/10.1016/j.infsof.2019.08.002.

Sánchez Reolid, R., García, A., Vicente-Querol, M., Fernández-Aguilar, L., López, M., González, A., 2018. Artificial neural networks to assess emotional states from brain-computer interface. Electronics 7, 384. http://dx.doi.org/10.3390/electronics7120384.

Scheirer, J., Fernandez, R., Klein, J., Picard, R.W., 2002. Frustrating the user on purpose: a step toward building an affective computer. Interact. Comput. 14, 93–118. http://dx.doi.org/10.1016/S0953-5438(01)00059-5.

Senior, C., Russell, T., Gazzaniga, M.S., 2009. Methods in mind. In: Cognitive Neuroscience. The MIT Press.

Sforza, C., Rango, M., Galante, D., Bresolin, N., Ferrario, V., 2008. Spontaneous blinking in healthy persons: An optoelectronic study of eyelid motion. Ophthalmic Physiol. Opt. J. Br. Coll. Ophthalmic Opt. Optom 28, 345–353. http://dx.doi.org/10.1111/j.1475-1313.2008.00577.x.

Shaffer, F., Ginsberg, J., 2017. An overview of heart rate variability metrics and norms. Front. Public Health 5, 258. http://dx.doi.org/10.3389/fpubh.2017.00258.

Shaffer, T.R., Wise, J.L., Walters, B.M., Müller, S.C., Falcone, M., Sharif, B., 2015. Itrace: enabling eye tracking on software artifacts within the IDE to support software engineering tasks. In: Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2015. Presented at the 2015 10th Joint Meeting. ACM Press, Bergamo, Italy, pp. 954–957. http://dx.doi.org/10.1145/2786805.2803188.

Sharaev, M., Zinchenko, V., Ushakov, V., Kartashov, S., Velichkovsky, B., 2016. Effective connectivity within the default mode network: Dynamic causal modeling of resting-state fMRI. Data. Front. Hum. Neurosci. 10. http://dx.doi.org/10.3389/fnhum.2016.00014.

Sharafi, Z., Shaffer, T., Sharif, B., Gueheneuc, Y.-G., 2015a. Eye-tracking metrics in software engineering. In: 2015 Asia-Pacific Software Engineering Conference (APSEC). Presented at the 2015 Asia-Pacific Software Engineering Conference (APSEC). IEEE, New Delhi, pp. 96–103. http://dx.doi.org/10.1109/APSEC.2015.53.

Sharafi, Z., Shaffer, T., Sharif, B., Gueheneuc, Y.G., 2016. Eye-tracking metrics in software engineering. In: Proc. - Asia-Pac. Softw. Eng. Conf. APSEC 2016-May. pp. 96–103. http://dx.doi.org/10.1109/APSEC.2015.53.

Sharafi, Z., Sharif, B., Guéhéneuc, Y.-G., Begel, A., Bednarik, R., Crosby, M., 2020. A practical guide on conducting eye tracking studies in software engineering. Empir. Softw. Eng. 20, 442. http://dx.doi.org/10.1007/s10664-020-09829-4.

Sharafi, Z., Soh, Z., Guéhéneuc, Y.-G., 2015b. A systematic literature review on the usage of eye-tracking in software engineering. Inf. Softw. Technol. 67, 79–107. http://dx.doi.org/10.1016/j.infsof.2015.06.008.

Sharif, B., Clark, B., Maletic, J., 2016. Studying developer gaze to empower software engineering research and practice. pp. 940–943. http://dx.doi.org/10.1145/2950290.2983988.

Sidorova, A., Evangelopoulos, N., Valacich, J.S., Ramakrishnan, T., 2006. Uncovering the intellectual core of the information systems discipline. MIS Q 32, 467–482.

Siegmund, J., Brechmann, A., Apel, S., Kästner, C., Liebig, J., Leich, T., Saake, G., 2012. Toward measuring program comprehension with functional magnetic resonance imaging. In: Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering - FSE '12. Presented at the ACM SIGSOFT 20th International Symposium. ACM Press, Cary, North Carolina, p. 1. http://dx.doi.org/10.1145/2393596.2393624.

Siegmund, J., Kästner, C., Apel, S., Parnin, C., Bethmann, A., Leich, T., Saake, G., Brechmann, A., 2014. Understanding understanding source code with functional magnetic resonance imaging. In: Proceedings of the 36th International Conference on Software Engineering - ICSE 2014. Presented at the 36th International Conference. ACM Press, Hyderabad, India, pp. 378–389. http://dx.doi.org/10.1145/2568225.2568252.

Siegmund, J., Peitek, N., Brechmann, A., Parnin, C., Apel, S., 2020. Studying programming in the neuroage: Just a crazy idea? Commun. ACM 63, 30–34. http://dx.doi.org/10.1145/3347093.

Siegmund, J., Peitek, N., Parnin, C., Apel, S., Hofmeister, J., Kästner, C., Begel, A., Bethmann, A., Brechmann, A., 2017. Measuring neural efficiency of program comprehension. In: Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2017. Presented at the 2017 11th Joint Meeting. ACM Press, Paderborn, Germany, pp. 140–150. http://dx.doi.org/10.1145/3106237.3106268.

Sommerville, I., 2010. Software Engineering, 9th ed. Addison-Wesley Publishing Company, USA.

Sweller, J., 2011. CHAPTER TWO - cognitive load theory. In: Mestre, J.P., Ross, B.H. (Eds.), Psychology of Learning and Motivation. Academic Press, pp. 37–76. http://dx.doi.org/10.1016/B978-0-12-387691-1.00002-8.

Tarasau, H., Thapaliya, A., Zufarova, O., 2019. Problems in experiment with biological signals in software engineering: The case of the EEG. In: Mazzara, M., Bruel, J.-M., Meyer, B., Petrenko, A. (Eds.), Proceedings of the 51st International Conference on Software Technology: Methods and Tools. Springer International Publishing, Cham, pp. 81–88.

Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology, 1996. Heart rate variability: Standards of measurement, physiological interpretation, and clinical use. Circulation 93, 1043–1065.

Thapaliya, A., 2019. EEG: identification of concentration level under pair programming. In: Kruglov, A.V., Aksyonov, K.A. (Eds.), Proceedings of the 6th International Young Scientists Conference on Information Technologies, Telecommunications and Control Systems (ITTCS 2019).

Toga, A.W., Mazziotta, J.C., 2002. Brain Mapping: The Methods, second ed. Academic Press.

Topi, H., Tucker, A. (Eds.), 2014. third ed. Computing Handbook, Third Edition: Information Systems and Information Technology, Computing Handbook, (2), CRC Press.

Vessey, I., Ramesh, V., Glass, R.L., 2002. Research in information systems: An empirical study of diversity in the discipline and its journals. J. Manag. Inf. Syst. 19, 129–174.

Villringer, A., Chance, B., 1997. Non-invasive optical spectroscopy and imaging of human brain function. Trends Neurosci. 20, 435–442. http://dx.doi.org/10.1016/S0166-2236(97)01132-6.

vom Brocke, J., Hevner, A., Léger, P., Walla, P., Riedl, R., 2020. Advancing a neurois research agenda with four areas of societal contributions. Eur. J. Inf. Syst. 29, 9–24. http://dx.doi.org/10.1080/0960085X.2019.1708218.

vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A., 2009. Reconstructing the giant: On the importance of rigour in documenting the literature search process. http://Www.Alex{and}ria.Unisg.Ch/Publikationen/67910.

Vrzakova, H., Begel, A., Mehtätalo, L., Bednarik, R., 2020. Affect recognition in code review: An in-situ biometric study of reviewer's affect. J. Syst. Softw. 159, 110434. http://dx.doi.org/10.1016/j.jss.2019.110434.

Walla, P., Nesbitt, K., Blackmore, K., Hookham, G., Kay-Lambkin, F., 2015. Using the startle eye-blink to measure affect in players. pp. 401–434. http://dx.doi.org/10.1007/978-3-319-05834-4_18.

Webster, J., Watson, R., 2002. Analyzing the past to prepare for the future: Writing a literature review. MIS Q. 26. http://dx.doi.org/10.2307/4132319.

Wel, P., Steenbergen, H., 2018. Pupil dilation as an index of effort in cognitive control tasks: A review. Psychon. Bull. Rev. 25. http://dx.doi.org/10.3758/s13423-018-1432-y.

Wise, J., Prox, B., Clark, B., Sharif, B., 2016. Invited talk: Towards an emotionally aware development environment. In: 2016 IEEE/ACM 1st International Workshop on Emotional Awareness in Software Engineering (SEmotion). pp. 26–27. http://dx.doi.org/10.1109/SEmotion.2016.015.

Wrobel, M., 2018. Applicability of emotion recognition and induction methods to study the behavior of programmers. Appl. Sci. 8, 323. http://dx.doi.org/10.3390/app8030323.

Wróbel, M., Wrobel, M.R., 2018. Methods and tools for affect-aware intelligent tutoring systems view project applicability of emotion recognition and induction methods to study the behavior of programmers. Appl. Sci. 8, 323. http://dx.doi.org/10.3390/app8030323.

Wulff-Jensen, A., Ruder, K., Triantafyllou, E., Bruni, L.E., 2019. Gaze strategies can reveal the impact of source code features on the cognitive load of novice programmers. In: Ayaz, H., Mazur, L. (Eds.), Advances in Neuroergonomics and Cognitive Engineering. Springer International Publishing, Cham, pp. 91–100.

Xu, J., Liu, X., Zhang, J., Li, Z., Wang, X., Fang, F., Niu, H., 2015. FC-NIRS: A functional connectivity analysis tool for near-infrared spectroscopy data. BioMed Res. Int. 2015, 1–11. http://dx.doi.org/10.1155/2015/248724.

Yamamoto, A., Uwano, H., Ikutani, Y., 2016. Programmer's electroencephalogram who found implementation strategy. In: 2016 4th Intl Conf on Applied Computing and Information Technology/3rd Intl Conf on Computational Science/Intelligence and Applied Informatics/1st Intl Conf on Big Data, Cloud Computing, Data Science & Engineering (ACIT-CSII-BCD). Presented at the 2016 4th Intl Conf on Applied Computing and Information Technology/3rd Intl Conf on Computational Science/Intelligence and Applied Informatics/1st Intl Conf on Big Data, Cloud Computing, Data Science & Engineering (ACIT-CSII-BCD). IEEE, Las Vegas, NV, USA, pp. 164–168. http://dx.doi.org/10.1109/ACIT-CSII-BCD.2016.041.

Yeh, M.K.-C., Gopstein, D., Yan, Y., Zhuang, Y., 2017. Detecting and comparing brain activity in short program comprehension using EEG. In: 2017 IEEE Frontiers in Education Conference (FIE). Presented at the 2017 IEEE Frontiers in Education Conference (FIE). IEEE, Indianapolis, IN, pp. 1–5. http://dx.doi.org/10.1109/FIE.2017.8190486.

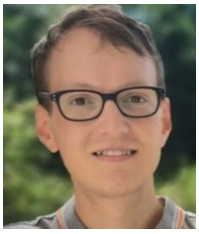
Zimeo Morais, G., Balardin, J., Sato, J., 2018. FNIRS Optodes' Location Decider (fOLD): A toolbox for probe arrangement guided by brain regions-of-interest. Sci. Rep. 8. http://dx.doi.org/10.1038/s41598-018-21716-z.

Züger, M., Fritz, T., 2015. 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 '15. Presented at the 33rd Annual ACM Conference. ACM Press, Seoul, Republic of Korea, pp. 2981–2990. http://dx.doi.org/10.1145/2702123.2702593.

Züger, M., Fritz, T., 2018. Sensing and supporting software developers' focus. In: Proceedings of the 26th Conference on Program Comprehension - ICPC '18. Presented at the 26th Conference. ACM Press, Gothenburg, Sweden, pp. 2–6. http://dx.doi.org/10.1145/3196321.3196323.

Züger, M., Müller, S.C., Meyer, A.N., Fritz, T., 2018. Sensing interruptibility in the office: A field study on the use of biometric and computer interaction sensors. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18. ACM, New York, NY, USA, pp. 591:1–591:14. http://dx.doi.org/10.1145/3173574.3174165.

Barbara Weber is Professor for Software Systems Programming and Development at the University of St. Gallen, Switzerland. She is Chair for Software Systems Programming and Development, Dean of the School of Computer Science, and Director of the Institute of Computer Science. Barbara's research interests include human and cognitive aspects in software and process engineering. Barbara currently has an h-index of 47 and has published more than 160 refereed papers, for example, in Nature Scientific Reports, Information and Software Technology, Software and System Modeling, Information Systems, Data and Knowledge Engineering, and Journal of Management Information systems.



Thomas Fischer is a university assistant at the department of Digital Business at the University of Linz. He received his doctorate in social sciences and economics from the University of Linz, Austria, in 2020. His research interests are in digital stress in organizations and agility in IS development projects.

René Riedl is professor of Digital Business and Innovation at the University of Applied Sciences Upper Austria and currently vice dean at the School of Business and Management. Also, he is an associate professor of Business Informatics at the University of Linz. Moreover, he is the scientific director of the NeuroIS Society and serves as a conference co-chair of the NeuroIS Retreat (www.NeuroIS.org). He has published among others in BMC Neurology, Business & Information Systems Engineering, European Journal of Information Systems, Journal of Information Technology, Journal of Management Information Systems, Journal of Neuroscience, Psychology, and Economics, Journal of the Association for Information Systems, PLoS ONE, and MIS Quarterly.