Journal of the Association for Information Systems

Research Paper ISSN: 1536-9323

Brownie: A Platform for Conducting NeuroIS Experiments

Anuja Hariharan
Karlsruhe Institute of Technology, Germany
anuja.hariharan@kit.edu

Marc Thomas Philipp Adam
The University of Newcastle, Australia

Verena Dorner
Karlsruhe Institute of Technology, Germany

Ewa Lux
Karlsruhe Institute of Technology, Germany

Marius B. Müller
Karlsruhe Institute of Technology, Germany

Jella Pfeiffer
Karlsruhe Institute of Technology, Germany

Christof Weinhardt
Karlsruhe Institute of Technology, Germany

Abstract:

In the NeuroIS field, experimental software needs to simultaneously present experimental stimuli to participants while recording, analyzing, or displaying neurophysiological measures. For example, a researcher might record a user's heart beat (neurophysiological measure) as the user interacts with an e-commerce website (stimulus) to track changes in user arousal or show a user's changing arousal levels during an exciting game. In this paper, we identify requirements for a NeuroIS experimental platform that we call Brownie and present its architecture and functionality. We then evaluate Brownie via a literature review and a case study that demonstrates Brownie's capability to meet the requirements in a complex research context. We also verify Brownie's usability via a quantitative study with prospective experimenters who implemented a test experiment in Brownie and an alternative software. We summarize the salient features of Brownie as follows: 1) it integrates neurophysiological measurements, 2) it incorporates real-time processing of neurophysiological data, 3i) it facilitates research on individual and group behavior in the lab, 4) it offers a large variety of options for presenting experimental stimuli, and 5) it is open source and easily extensible with open source libraries. In summary, we conclude that Brownie is innovative in its potential to reduce barriers for IS researchers by fostering replicability and research collaboration and to support NeuroIS and interdisciplinary research in cognate areas, such as management, economics, or human-computer interaction.

Keywords: NeuroIS, Experimental Software, Behavioral Research.

Varun Grover was the accepting senior editor. This paper was submitted on June 16, 2015, and went through three revisions.

1 Introduction

In recent years, the proliferation of built-in biosensors has made it possible to integrate biosignals as real-time system input for everyday use, such as in stress management (Riedl, 2013) and emotion regulation for financial trading (Djajadiningrat, Geurts, Munniksma, Christiaansen, & de Bont, 2009; Astor, Adam, Jerčić, Schaaff, & Weinhardt, 2013). As a subfield of IS research, the area of NeuroIS builds on the advances in biosensor technology and applies neuroscience theories, methods, and tools to contribute to: 1) the development of theories that enable more accurate predictions and explanations of IT-related behaviors and 2i) the design of IT artifacts, which positively influence technology adoption and user experience (Dimoka, Pavlou, & Davis, 2011; Riedl et al., 2010a; Riedl, Davis, & Hevner, 2014a).

Neurophysiological data contributes to our better understanding constructs such as emotions (Gregor, Lin, Gedeon, Riaz, & Zhu, 2014), cognitive load (Ortiz de Guinea, Titah, & Léger, 2013), or trust (Riedl, Mohr, Kenning, Davis, & Heekeren, 2014b), which are important predictors of IT-related behavior. Researchers have applied NeuroIS methods in the domains of assistive technologies (Randolph & Jackson, 2010), work-related IS use patterns (Ortiz de Guinea & Webster, 2013), seller and buyer behavior (Randolph, Borders, & Loe, 2013; Riedl et al., 2014b; Adam, Krämer, Jähnig, Seifert, & Weinhardt, 2011), serious games (Li, Jiang, Tan, & Wei, 2014; Jerčić et al., 2012), recommender systems (Pfeiffer, Pfeiffer, & Meißner, 2015), technostress (Riedl, 2013), and user acceptance (Kjærgaard & Jensen, 2014). Beyond informing the design of IT artifacts, NeuroIS also enables one to develop neuro-adaptive information systems; that is, "systems that recognize the physiological state of the user and that adapt, based on that information, in real-time" (Riedl et al., 2014a, p. i).

Researchers have conducted the majority of NeuroIS studies within the scope of laboratory experiments to study individual and group IS phenomena while establishing high levels of control (vom Brocke & Liang, 2014; vom Brocke, Riedl, & Léger, 2013). Indeed, researchers in NeuroIS and other cognate areas such as behavioral economics, neuro-economics, and affective computing have recognized the need for such control (Shim, Varshney, & Dekleva, 2006). These approaches require innovative and accessible methodological toolsets to support researchers in developing a deeper understanding of how they enrich human-computer interaction and human decision making.

Establishing freely accessible toolsets that address researchers' needs aids the NeuroIS community and closely related areas in several ways. In particular, such tools can be instrumental in 1) reducing the barriers for IS researchers to engage in collaborative NeuroIS research, 2) increasing the comparability and documentation across studies, and 3) increasing the replicability of studies. Prominent examples for software tools that have aided the research communities of behavioral experimental research in general and NeuroIS research in particular include z-Tree by Fischbacher (2007) (which became a workhorse in experimental economics with about 6400 citations), Ledalab by Benedek and Kaernbach (2010) for analyzing electrodermal activity (about 200 citations), ERPSim by Léger (2006) (about 140 citations), ORSEE by Greiner (2004) for recruiting participants (about 1600 citations), and PhysioNet by Moody, Mark, and Goldberger (2001) for analyzing physiological data (about 200 citations)[1]. But while current software tools are well suited to solving issues in specific domains, we recognize a research gap for a platform for conducting NeuroIS experiments that can be used across several domains, facilitates sensor-data collection, performs real-time biosignal analyses, works in both individual and group interactive research scenarios, and, finally, is open source and fosters collaboration.

To overcome these challenges and to enable flexible integration of NeuroIS tools into experimental IS research, we introduce Brownie (behavioral research of individuals and groups using Web and NeuroIS experiments). We began the development process of the platform three years ago and iteratively defined the requirements and enhanced its features via experiments, use cases, and workshops. Brownie facilitates NeuroIS research in the lab by enabling one to collect, store, and synchronize sensor-data events (such as electrocardiography (ECG), electrodermal activity (EDA), photoplethysmography (PPG), electro-encephalography (EEG)). In addition, Brownie is geared towards integrating the real-time processing of neurophysiological measurements, which enables research in upcoming areas such as neuro-adaptive systems. Finally, beyond its capabilities for NeuroIS research, Brownie supports research in interaction scenarios (between individuals or groups), which many IS contexts and group decision experiments require.

At present, the effort and technical knowledge required to conduct NeuroIS research are substantial. Hence, a freely accessible platform for conducting NeuroIS experiments could be instrumental in

[1] Results from Google Scholar search as of 18 August, 2016.

increasing transparency and fostering the exchange of research know-how across cognate areas. In summary, Brownie is innovative in that it reduces barriers for IS researchers to engage in collaborative research and is interdisciplinary in its potential to be a platform that increases the replicability of studies across disciplines of neurosciences, IS, economics, and management.

This paper proceeds as follows: in Section 2, we identify the need for a platform to facilitate NeuroIS research in the lab and outline the requirements for a potential solution platform. In Section 3, we discuss the design process we used to develop Brownie and its architecture in three iterations. In Section 4, we evaluate the platform via a literature review, a case study, and a usability study. In Section 5, we discuss the contributions and limitations of this work and provide directions for future research. Finally, in Section 6, we conclude the paper.

2 Problem Identification and Requirements Definition

As part of the IS research field, NeuroIS research broadly deals with the same objectives of investigation as the IS field in general: designing and understanding information systems and users' interactions with them. As Dimoka et al. (2012, p. 680) state: NeuroIS promises "to complement existing research tools with neurophysiological tools that can provide reliable data which are difficult or impossible to obtain with traditional tools, such as self-reported or archival data". NeuroIS shares an interest in understanding and improving human-computer interaction with related areas such as psychology and behavioral economics. NeuroIS also faces difficulties that arise in studying such questions—in particular, the potential complexity of the interaction scenario with respect to participant, situation, and system-specific factors that might confound study results. For NeuroIS, other factors further compound the problem such as the difficulties inherent in measuring and processing several modalities simultaneously and spatial and temporal resolution in measurements (Riedl & Léger, 2016, p. 47). These aspects require one to develop controlled experiments to understand system design and its effects on behavior. To this end, we formulate the need for experimentation in human-computer interaction based systems as a central problem that NeuroIS researchers face (Gregor et al., 2014; Ortiz de Guinea et al., 2013).

Hence, in this paper, we address this central problem by defining a solution platform's scope, providing a solution artifact, and evaluating the platform for this problem space. In the context of this paper, we define a platform as an experimental software that enables researchers to develop and conduct experiments by providing a foundation in terms of specialized libraries and interfaces on which they can build programs for specific research investigations and integrate third-party software components[2]. Such a platform can 1) provide standards that experiments need as the foundation 2) that one can extend via third-party NeuroIS components where necessary. To build and evaluate Brownie, we adopted the steps for a design science methodology as Peffers, Tuunanen, Rothenberger, and Charrerjee (2007) outline.

To define Brownie's scope, we first identified the experimental software that experimental laboratories in universities around the world use (behavioral, with and without sensor data) by emailing these laboratories and asking which software they used regularly for their behavioral experiments (NeuroIS and non-NeuroIS experiments). We contacted 46 laboratories in total, and 17 replied with information about the most commonly used software features in their respective laboratories. We summarize commonly used experimental software in Figure 1[3].

By surveying features from the manuals and websites of the commonly used experimental software, we formulated a set of requirements for NeuroIS experiments. Since experimental software included a wide range of features, we categorized them into four broad requirements: 1) individual and group interaction, 2) biosensors, 3) technical, and 4) auxiliary requirements. Figure 1 indicates what software satisfies what requirements. A literature review of NeuroIS studies (see Section 4) added two requirements to our list: 5) integration of questionnaires and 6) integration of multimedia. We formulated several auxiliary requirements (such as open source, common programming language, linking to a database, etc.) based

[2] Note that not all experimental software provides the integration capabilities of platforms such as Brownie, Noldus Observer, or Imotion. Some experimental software focuses on stimulus presentation such as the presentation software E-Prime and Presentation. Others offer toolboxes to conduct experiments such as the z-Tree toolbox within functionalities of existing libraries but do not allow one to create new libraries and integrate other software components. By contrast, a platform such as Noldus Observer, which is an interesting alternative to Brownie for a variety of settings, offers a wide range of integration functionalities as demonstrated in the experiments by Léger et al. (2014) and Charland et al. (2015).
[3] The list of experimental software in Figure 1 is based on the feedback from experimental laboratories around the world, discussions with NeuroIS researchers in focus groups and workshops, and the review process.

on informal discussions with NeuroIS scholars at conferences and workshops. We circulated the four requirement categories and the requirement list among and discussed them with scholars of NeuroIS and cognate domains at several workshops; notably, the 2014 NeuroIS Gmunden Retreat attended by eminent NeuroIS scholars, who confirmed that these requirements indeed represented common requirements and commonly faced issues while implementing NeuroIS experiments.

2.1 Requirement 1 (R1): Group & Individual Interactions

Experimental research in the IS and management domains is, to a large extent, devoted to studying users' decision making, perceptions, and behavior. From the perspective of NeuroIS, researchers are often interested in how users engage with the system (such as processing information, performing a cognitively demanding task, or making an important financial decision). Examining questions pertaining to human-computer interaction, studying, and measuring neurophysiological processes at the same time in these scenarios (Teubner, Adam, & Riordan, 2015) requires a solution platform capable in the above respects.

			Requirement 1 Individuals & Group		Requirement 2 Biosignals				Requirement 3 Technical Requirements					Requirement 4 Auxiliary Requirements			
No.	Platform	Year of latest version	Individuals (R1a)	Groups (R1b)	Biosignals (R2a)	Event synchronization (R2a)	Signal quality checks (R2b)	Real-time signal processing (R2c)	Support research on websites (R3a)	Flexibility of data logging (R3b)	Common programming language (R3c)	Extensible (R3d)	Multimedia (R3e)	Open source (R4a)	Tutorials and support (R4b)	Redistribution and replication (R4c)	Questionnaires and survey forms (R4d)
1	BoXS	2013	•	•					•				•	•	•	•	•
2	ConG	2013	•	•							•	•	•	•	•	•	
3	DirectRT	2014	•			•				•			•		•		•
4	EconPort	2006	•	•					•						•	•	
5	E-Prime 2	2008	•		•		•						•		•		
6	Imotions	2016	•		•	•	•		•	•		•	•		•		•
7	Inquisit	2014	•		•	•			•				•		•		•
8	JessX	2010	•	•							•			•	•	•	
9	MediaLab	2014	•										•		•		•
10	Noldus Observer	2015	•	•	•	•	•	•	•	•	•	•	•		•		•
11	Presentation	2016	•		•		•		•	•		•	•		•		
12	PsychoPy	2014	•			•					•	•	•	•	•	•	
13	Psychtoolbox	2013	•		•	•									•	•	
14	Regate	2009	•	•											•	•	•
15	Seaweed	2011	•	•					•	•	•		•			•	
16	SoPHIE	2014	•	•					•				•		•	•	•
17	STIM2	2014	•		•	•							•				
18	Superlab	2013	•		•		•						•		•		
19	VeconLab	2014	•	•					•							•	
20	z-Tree	2013	•	•						•			•		•	•	•
	Brownie	2016	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

Figure 1. Requirements Met by Current Experimental Software[4]

In addition, research in the NeuroIS domain seeks to understand individual and group behavior based on a person's internal state. The rise of Web 2.0 has facilitated group interaction in virtual environments, such as online auctions, collaboration for content generation, and multi-user digital gaming (Teubner et al.,

[4] Note: 1) Seithe (2012), 2) Pettit, Friedman, Kephart, & Oprea (2014), 3) Jarvis (2003), 4) Cox & Swarthout (2005), 5) Schneider, Eschman, & Zuccolotto (2002), 6) iMotions (2016), 7) Draine (1998), 8) JessX (2010), 9) Jarvis (2004), 10) Noldus (1991), 11) Presentation (2016), 12) Peirce (2007), 13) Brainard (1997), 14) Zeiliger (2000), 15) Chilton, Sims, Goldman, Little, & Miller (2009), 16) Hendriks (2012), 17) Neuroscan (2008), 18) Haxby, PArasuraman, Lalonde, & Abboud (1993), 19) Bostian & Holt (2013), 20) Fischbacher (2007). Software is listed in alphabetical order.

2015; Kuan, Zhong, & Chau, 2014). Research on the interaction of humans in groups in a technological environment (in both static and dynamic contexts[5]) continues to increase in scope and importance. As such, we formulate a NeuroIS platform's first requirement as follows:

R1: A NeuroIS platform needs to facilitate individual interaction (R1a) and group interaction (R1b) with systems.

2.2 R2: Biosignals

Although current experimental software provides capabilities such as stimulus presentation, human-computer interaction, or group interaction, most software is limited in ways that compromise their usability for NeuroIS research (most notably, real-time biosignal processing). NeuroIS research relies on continuously recording physiological data in order to better understand internal processes, underlying externally visible human behavior, such as activities and decisions, recorded as text inputs, mouse clicks (Dimoka, 2010; Schaaff, Degen, Adler, & Adam, 2012) or eye movements (Pfeiffer et al, 2014a; Pfeiffer, Meißner, Prosiegel, & Pfeiffer, 2014b). Employing biosignal data using sensors is an emerging method for assessing users' internal states (such as emotion, cognitive load, focus level, or relaxation state) that focuses on improving our understanding of human behavior and building user-centric information systems (Astor et al., 2013).

Several experimental software programs facilitate neurophysiological measurements with stimuli-based experiments in which individuals interact with a system. However, these software programs were not designed to integrate stimuli and events, which becomes increasingly complex when one acquires data from more than one sensor. In the context of multiple sensor modalities, Léger et al. (2014) provide guidelines for synchronizing sensor data from multiple sources. In the case they describe, multiple recording devices (one for each sensor) record and store sensor data separately, each based on an individual clock. In order to later synchronize the timestamps of the different clocks, Léger et al. (2014) propose an additional device that has the dedicated task of sending marker signals to all connected recording devices. One then later uses these marker signals when post-processing the data as reference points for synchronization. Hence, based on prior NeuroIS guidelines, we conclude that a NeuroIS platform requires temporal synchronization of multiple modalities and synchronization with events in the experiment. As such, we formulate a NeuroIS platform's second requirement as follows:

R2(a/b): A NeuroIS platform needs the ability to store unimodal and multimodal biosensor data and temporally synchronize these data with the various events in the experiments in a uniform and ready-to-analyze format (R2a). Further, such a platform needs the ability to perform signal quality checks, to confirm whether biosensors are connected, and to verify whether biosensor data is being continuously stored (R2b).

An emerging requirement of NeuroIS research is the ability to integrate the real-time processing of biosignals and further design systems based on this information. Researchers use user input in the form of real-time biosignal data to adapt the system response or system design in real-time according to the user's cognitive or affective state of the user (see strategy 3, vom Brocke et al., 2013). As Riedl (2013, p. 44) states in his research agenda, "Design science researchers could contribute to the development of information systems, which use bio-signals as real-time system input in order to make human-computer interaction less stressful, and hence more convenient, enjoyable, and effective". Researchers have termed these systems as "neuro-adaptive" (Riedl et al., 2014a; Riedl & Léger, 2016) and developed them to improve users' experience, reduce users' stress perceptions, or help users achieve certain goals. One specific example of a neuro-adaptive system feature is the live-biofeedback, which informs users (based on their biosignals processed in real time) about their current internal state. Researchers have used live-biofeedback in serious games and technology-enhanced learning environments (Astor et al., 2013; Ouwerkerk et al., 2013). In summary, neuro-adaptive systems mandate the need to integrate real-time signal processing features in an experimental platform. As such, we extend the second requirement as follows:

R2(c): A NeuroIS platform needs the ability to incorporate features for real-time processing of sensor data in order to design neuro-adaptive systems (e.g., interface adaptation, interventions, live-biofeedback) for experimentation.

[5] Static refers to round-based settings wherein participants wait for others in the group to complete their actions before they can observe them. Dynamic scenarios refer to those where one participant observes the actions of other participants in real time.

2.3 R3: Technical Requirements

We next identify a list of general technical requirements that are common to the majority of IS experiments. IS experiments often study users' behavior while they browse website content or make buy/sell decisions on e-commerce platforms (Dorner, Ivanova, & Scholz, 2013; Gregor et al., 2014). From a NeuroIS perspective, enriching users' click and search pattern data on websites with physiological measurements taken in a controlled lab environment opens up the possibility of connecting user behavior with internal states in order to be able to understand their behavior and reaction to system design changes (see Dimoka, 2010; Fadel, Meservy, & Jensen, 2015; Minas, Potter, Dennis, Bartelt, & Bae, 2014). As such, we formulate a NeuroIS platform's third requirement as follows:

> **R3(a)**: A NeuroIS platform needs to allow one to conduct IS experiments on websites in a controlled lab environment.

Tracking and logging of user data and activities is an important element in analyzing user behavior. Primarily, flexibility regarding the type of information that can be stored (such as different inputs or choices made by users in an interface, tracking button clicks, time of displaying certain information, temporal synchronization of events and multiple modalities, etc.) remains an important requirement in facilitating future analysis, potentially combining neurophysiological activity (Léger et al., 2014; Hu, West, & Smarandescu, 2015; Vance, Anderson, Kirwan, & Eargle, 2014) as noted above. As such, we extend the third requirement as follows:

> **R3(b)**: A NeuroIS platform needs the ability to provide flexible user activity logging (in terms of types and format of information) for experimenters.

Another general technical requirement across experimental software is the ease-of-use in implementing experiments with potentially complex interfaces and logic. This can be achieved by allowing programming in a commonly used programming language with adequate implementation support, for example on forums and in books (Kuan et al., 2014; Li et al., 2014). As such, we extend the third requirement as follows:

> **R3(c)**: A NeuroIS platform needs the ability to allow one to program an experiment with a commonly used programming language and the ability to implement customized procedures and interfaces in the lab.

Interlinked with the previous requirement of a common programming language, is the scalability and extensibility aspect of the platform. While most of the experimental software surveyed in Figure 1 are able to integrate a specified list of devices, they are not extensible or scalable to be combined with other devices, such as mobile gadgets. As such, we extend the third requirement as follows:

> **R3(d)**: A NeuroIS platform needs to allow one to extend it via available libraries for biosensor and hardware requirements.

Finally, in order to measure the underlying physiology, stimuli have been provided in the form of text (Minas et al., 2014), images (Benbasat, Dimoka, Pavlou, & Qiu, 2010; Gregor et al., 2014; Riedl et al., 2014b), sounds (Jerčić et al., 2012), or videos (Nogueira, Aguiar, Rodrigues, & Oliveira, 2014). As such, we extend the third requirement as follows:

> **R3(e)**: A NeuroIS platform needs to allow one to incorporate various multimedia elements as stimuli in experiment interfaces.

2.4 R4: Auxiliary Requirements

In terms of the advancement that prior NeuroIS research findings have created, the distribution model one uses represents an important aspect of a NeuroIS platform. In surveying current software, we found two kinds of distribution models: open source models that allow one to modify their source code and models that neither distribute the source code nor allow one to modify it. More importantly, distribution models effect different advantages and disadvantages for the researchers who want to carry out experiments. For example, commercially available software is commonly not open source but provides various support services for experimenters. By contrast, open source software is often limited in terms of support services and accountability but achieves high levels of transparency of implementation details and facilitates the exchange of knowledge. Based on the feedback from the initial requirement-gathering phase, researchers stated open source software as their preferred choice because it helps other researchers replicate NeuroIS experiments by sharing the source code, enables them to extend the platform for new experiments, and fosters collaboration among researchers (e.g., by jointly developing libraries for the platform). As such, we formulate a NeuroIS platform's fourth requirement as follows:

> **R4(a)**: A NeuroIS platform needs to be open source.

In order to help researchers easily and continuously use a platform, the platform needs to provide a sufficient number of tutorials and use cases and documentation and support via issue trackers or forums, which will encourage discussion on research's use and extensibility. As such, we extend the fourth requirement as follows:

> **R4(b)**: A NeuroIS platform needs to provide tutorials, documentation, and support methods.

Finally, one needs to be able to replicate and reuse behavioral research conducted in a lab (Friedman & Sunder, 1994). Riedl and Léger (2016, p. 112) state that, for NeuroIS experiments, "[the] possibility of replication is an important pre-condition of a study's objectivity". As such, we extend the fourth requirement as follows:

> **R4(c)**: A NeuroIS platform needs to allow one to incorporate new functionalities into existing experiments and allow for distribution and replication of existing experiments.

Collecting additional information about participants is a basic requirement of virtually all NeuroIS experiments and includes assessing additional data such as demographic information and specific personality traits or recording participants' perceptions during the experiment. As such, we extend the fourth requirement as follows:

> **R4(d)**: A NeuroIS platform needs to allow one to incorporate questionnaires in a controlled lab environment.

3 Design, Development, and Demonstration

We initially began to develop Brownie when we became aware of the growing complexity and technological challenges researchers face in conducting NeuroIS experiments, which result in part due to experimental software's limits in terms of extensibility, group interaction, and biosignal integration. Hence, we envisioned a staged plan to iteratively design a platform that would address those needs. In this section, we outline the iterative problem-solving process used to design and develop Brownie. We applied a design as a search process approach to decompose the complexity of developing a platform for laboratory experiments in the domain of NeuroIS into subproblems as Peffers et al. (2007) and Hevner, March, Park, and Ram (2004) recommend. The four requirements we outline above represent these subproblems, which we addressed in three main iterations of design and development. Six use cases (i.e., experiments by internal and external researchers) guided the design and development process to demonstrate the feasibility of the requirements and evaluate the platform version developed in each iteration.

3.1 Iteration 1: Basic Client-server Architecture

In the first iteration, we implemented Brownie's core client-server structure. This first implementation featured individual (R1a) and group (R1b) interactions. Brownie is Java-based (R3c), includes flexible data logging (R3b), and its layered architecture retains modularity and, hence, extensibility (R3d) while abstracting functionality. We chose Java was as the language of implementation since several operating systems support it and many applications, enterprise software, and e-commerce solutions widely use it. Finally, Java is the preferred choice of language in the case of extensions to mobile development due to its compatibility with the Android system.

One can classify architecture components along two dimensions: 1) whether they are part of the client or the server side and 2) whether they belong to the built-in or the customizable tier. In order to ensure that the platform encapsulates components and abstracts information (e.g. message passing methods, database mapping) about the layers below the layer exposed to experimenters, we separated the platform into a built-in tier (the core) and a customizable tier for experimenters. The built-in tier is ready to use and requires no alterations when implementing an experiment. Figures A1 and A2 visualize the built-in tier. The customizable tier, on the other hand, makes it possible to implement new experiments. Experimenters can specify user interfaces for clients, specify grouping rules for subjects in various periods, assign roles to clients, and customize the experimental flow. Exemplary experiments are distributed with the source code to help first-time Brownie users more easily use it.

According to design as a search process method (Peffers et al., 2007; Hevner et al., 2004), each iteration of the creative and heuristic problem solving process must produce a representation of an artifact that is being demonstrated. With respect to this first iteration of designing and developing Brownie, we created artifacts for two standard decision scenarios: trust game (Riedl et al., 2014b) and ultimatum game (Joe & Lin, 2008; Loewenstein, 2001). We used both games to confirm the operability of Brownie's basic architecture. The games incorporated basic solutions for managing roles, grouping subjects, and handling errors and exceptions. Demonstrating and discussing these artifacts revealed several ways in which we needed to adapt the platform for advanced experimentation, which we addressed at the end of this iteration. For instance, the ultimatum game had two roles and a role-matching requirement, but other experiments do not necessarily require this step; hence, we implemented a default single-role-matcher that assigned all subjects to the same role throughout an experiment. We also facilitated custom and experiment-specific role matching methods in Brownie via abstract classes. Similarly, we identified general scenarios for grouping (such as partner matching, random matching, perfect stranger matching, etc.) and implemented them.

3.2 Iteration 2: Integrating Sensors for Biosignal Acquisition

The second iteration addressed the acquisition of biosignals (R2a), including signal quality checks (R2b) and the real-time processing of biosignal data (R2c). The biosensor management layer allows one to set up, connect, and record biosensor data via Brownie. For this purpose, we incorporated a standardized biosensor interface with basic functions such as starting/stopping recording and specifying locations for saving physiological data into Brownie. One can extend the standardized interface to specify and configure sensor-specific properties (such as sampling frequency, hardware connectivity information for parallel port, or Bluetooth properties).

In the most recent iteration, we implemented Brownie with the ability to record ECG, EDA, PPG, EEG, eye-tracking data, and audio and webcam data. As for real-time processing, Brownie provides a generic extensible interface that helps one to specify sensors' configuration, handle data streams, and specify real-time parameters (e.g., sampling frequency, time window for real-time computation). Brownie provides real-time processing of heart rate and skin conductance using ECG and EDA measurements, which one can use to adjust user interface components based on a user's affective state (e.g., live biofeedback). The modularity of the biosensor tier allows one to implement new real-time monitoring and analyses methods (i.e. filtering methods, processing algorithms, and post-processing algorithms).

Brownie addresses data synchronization in two different ways: integrated or offline. In the first way (integrated), Brownie records all biosensor modalities itself. Hence, Brownie timestamps all data using the same clock and one does not need to conduct any post hoc synchronization. As such, for most experiments, Brownie records all biosensor modalities with one sensor device that is associated with one experimental computer. In the second way (offline), Brownie does not record at least one biosensor modality directly and one has to synchronize data offline. Here, one uses Brownie to record the behavioral data of the experiment, and it simultaneously acts as a marker signal emitter for all connected biosensors (cf. Hariharan, Adam, & Fuong, 2014). One can then use the marker signals for post-processing synchronization of one or more modalities. In addition to those two data-synchronization methods, Brownie always records timestamps for experiment events with server and client time in order to allow one to synchronize multiple clients with the server time post hoc. These data-synchronization capabilities are in the now-established NeuroIS guidelines that Dimoka et al. (2012) and Léger et al. (2014) raise (e.g., simultaneously obtaining data from different sources and providing a marker signal emitter functionality). In order to demonstrate the operability of the second iteration artifact (namely, the biosensor tier), we conducted three experiments (Appendix B). The first two experiments involved collecting large amounts of data (e.g., up to two hours of measurement for nine clients at a sample rate of 1000 Hz with three different sensors simultaneously), while experiment 3 included real-time signal processing. All three experiments verified the feasibility of collecting physiological data using Brownie. In this phase, the challenges we faced include those that concerned: 1) Bluetooth driver compatibility, 2) Java versioning and Dynamic Link Library (DLL) compatibility with various operating systems and sensor devices, and 3) the identification of processing speed requirements for real-time signal processing. We later identified and accounted for these challenges prior to the experiment setup and, subsequently, integrated appropriate solutions in Brownie.

3.3 Iteration 3: Preparation for Open Source Usage

In the third iteration, we prepared Brownie and its support landscape for open source use in the NeuroIS community, which included open source distribution (R3b), a support infrastructure of instructions, tutorials, and exemplary experiments (R4b), and the ability to redistribute and replicate implemented experiments (R4c). In addition, the third iteration addressed further requirements we identified in our literature review (see Section 4), in discussions with other experimenters (see Section 2), and in iterations 1 and 2, such as support for website research (R3a), integration of multimedia content (R3e), and integration of questionnaires (R4d).

Brownie is fully open source and licensed under the Apache 2.0 open source license with the added requirement of a citation in case of academic use. The support infrastructure includes a wiki with information about Brownie's architecture, basic usage instructions, and a FAQ section. It further includes several video tutorials that demonstrate how to set up existing experiments and how to implement a new experiment on Brownie. Other video tutorials address specific questions that experimenters commonly ask (e.g., how to install JWindow Builder, how data logging on Brownie works, and how to match subjects to different treatments). With respect to redistribution, the database engine PL/SQL proved difficult to install separately. Hence, we switched to the database engine H2, which dynamically creates a schema without requiring the experimenter to separately create it. Users could then add new values for the database to store with minimal recreation efforts. Further, to further allow researchers to distribute and replicate experiments, we made Brownie available on Bitbucket and gave all experimenters the opportunity to share their experimental code in the Bitbucket repository. We tested the result of the third iteration, a software artifact including its documentation and support infrastructure, in use cases 4, 5 and 6 (Appendix B). External experiments created the experiments described in these use cases based on the documentation and support material.

4 Evaluation

To evaluate Brownie, we adopted a combination of evaluation methods as Peffers et al. (2007) suggest. First, we analyzed the literature to confirm that Brownie meets the requirements of current experiments in the NeuroIS field, which demonstrates its utility in a broad domain. Second, we conducted a case study to evaluate Brownie's capability to meet the specific requirements in a complex NeuroIS research context. Third, we conducted a quantitative experimental study to examine the usability of Brownie in which participants implemented an experiment on Brownie and another experimental software and then assessed the respective usability. The three methods evaluated the criteria of "fulfillment of requirements" and "usability". The first two methods evaluated the fulfillment of requirements on a three-point scale (fulfilled, not fulfilled, not required in a particular study). The second and third methods evaluated usability on the dimensions of system usefulness, information quality, and interface quality by analyzing the qualitative case study data (interviews, documents, etc.) and the quantitative data from participants' responses to questions from the IBM usability scales (Lewis, 1995), respectively.

4.1 Evaluation of Requirements Based on a Literature Review

First, we thoroughly reviewed the NeuroIS research in the Senior Scholars' basket of eight journals that conducted an experiment with neurophysiological measurements to investigate their research questions, which yielded a total of 18 papers. To ensure independent assessment, an evaluator who was not a part of the developer team identified the requirements of the NeuroIS experiments conducted in these papers and validated whether Brownie met these requirements or, if that was not the case, if one could extend it to do so. We then contacted the author teams of the identified studies to confirm the assessment and revise where necessary (16 of the 18 author teams responded). Table 1 presents the results of the literature review. Requirements marked as "fulfilled" are met by Brownie and have already been demonstrated as feasible in one or more experiments. Requirements marked as "requiring extension" are technically feasible but have not been demonstrated via experiments yet. The literature review suggests that one can implement Brownie for nearly all published NeuroIS experiments. At this stage, four of the experiments (Dimoka, 2010; Riedl, Hubert, & Kenning, 2010b; Riedl et al., 2014b; Warkentin, Walden, Johnston, & Straub, 2016) would either require one to conduct offline analysis (fMRI measurements are commonly carried out by fMRI software, and event synchronization with experimental software is carried out offline) or to extend Brownie to access fMRI libraries. Moreover, one experiment (Léger et al., 2014) would require one to extend Brownie's real-time interface to process EEG data.

During this evaluation step, we identified two requirements commonly mentioned in the surveyed NeuroIS papers: integration of multimedia and integration of questionnaires. We implemented both requirements in Brownie, but we did not explicitly note them as requirements in the original requirement list. Since the literature review showed them to be important across several NeuroIS papers, we extended the requirement list accordingly. This evaluation also highlighted that the requirement list for the platform reflected the features required for conducting state-of-the-art NeuroIS experiments and would expand and change with the development of NeuroIS research.

Notably, at this stage, only two NeuroIS studies (Astor et al., 2013; Léger et al., 2014) in the Senior Scholars' basket of eight journals required real-time biosignal processing (R2c; see Table 1). However, eminent NeuroIS scholars have identified the integration of "neuroscience tools as built-in functions of IT artifacts" (vom Brocke et al., 2013, p. 3) as an important application area for NeuroIS that can lead to the development of "neuro-adaptive information systems" (Riedl et al., 2014a, p. xxix). Hence, we expect that, in the future, more NeuroIS will integrate biosignals as real-time systems input for, for example, real-time quality checks, the reduction of users' technostress perceptions, biofeedback applications, and brain-computer interfacing (Riedl & Léger, 2016, pp. 17-19).

4.2 Evaluation of Requirements and Usability Based on a Case Study

4.2.1 Case Study Description

We conducted our case study with two goals: 1) to investigate whether Brownie meets the requirements of experimenters and 2) to investigate whether it achieves high usability for experimenters and the participants of the experiment. The case study describes a research project that begun in March 2015, used Brownie, and whose structure broadly adopted the six essential phases of the NeuroIS research framework that vom Brocke and Liang (2014) introduce. No member of Brownie's developer team took part in this project in any capacity. The project team contacted the developer team a few times asking for access to the software and inquiring about tutorials and support material for specific questions (see Section 4.2.2). Two chief investigators (referred to as CI1 and CI2) planned to investigate user engagement on online participation platforms by comparing different crowdfunding mechanisms for funding four different projects. CI1 had experience in programming with z-Tree, an alternative to Brownie (see Figure 1), while CI2 had minimal prior experience in programming (programming courses only with no experience in programming projects). Both CIs had experience in experimental research. The CIs worked with two student research assistants (RA1 and RA2) on implementing their research project.

Table 2 overviews the data collected from the key members involved in this project during one or more of the above mentioned six phases. We interviewed two participants of the experiment who we randomly drew from the participant pool. In total, the research project lasted approximately six months, including all six research phases. Conducting the first two phases took up about two months (phase 1 and 2), designing and implementing the experiment in Brownie took about two-and-a-half months (phase 3), and conducting the experiment and analyzing the data took about one-and-a-half months (phases 4-6).

Table 1. Mapping of NeuroIS Experiments to Requirements (Paper Information)

Authors (year)	Outlet	Description	Major challenges	NeuroIS method	Software used	N
Astor et al. (2013)	JMIS	A biofeedback-based serious game for training emotion regulation skills.	Real-time biosignals to adapt the game to the user's physiological state.	HR	xAffect	104
Cyr, Head, Larios, & Pan (2009)	MISQ	An experiment to examine responses to website images.	Design of treatments; Calibration & correction.	Eye-tracking	Gaze-tracker	90
Dimoka (2010)	MISQ	An experiment to examine neural correlates of trust and distrust.	Stimuli manipulation; Correction of artifacts.	fMRI	SPM5	192
Fadel et al. (2015)*	JMIS	An experiment to study knowledge filtering process in online forums.	To orient subjects to the eye-tracking instrument; reduce novelty effects.	Eye-tracking	Tobii Studio	62
Gregor et al. (2014)	JMIS	A multi-method experiment of the nomological emotion network.	Synchronizing EEG equipment data to subjects' website viewing activity.	EEG	EEGLab	62
Hu et al. (2015)	JMIS	An experiment to study decision-making in information security.	Paradigm design; Recording and correction of EEG, ERP computation.	EEG	E-Prime	61
Kuan et al. (2014)	JMIS	An experiment to study opinions & emotions in group-buying	Group treatment manipulation, EEG data capture.	EEG	Emotiv Testbench/ EEGLab	18
Léger et al. (2014)	JAIS	The experiment to study neural reactions of users in a natural use context.	Reduce the artifacts; synchronize EEG and eye-tracking data.	EEG/ Eye-tracking	Net Station/ Tobii Studio/ Noldus Observer	24
Li et al. (2014)	JMIS	A study on how game elements impact user engagement.	Record and interpret the EEG data from the gaming process.	EEG	Emotiv Testbench/ EEGLab	44
Minas et al. (2014)	JMIS	Responses of subjects to new information during a text-based ICT discussion.	Multi-modal data to investigate subjects' cognitive and emotional responses.	EEG SC EMG	Emotiv Testbench/ MediaLab	44
Ortiz de Guinea & Webster (2013)	MISQ	A study to examine influence of events on IT use patterns.	Duration of experiment; Data from multiple modalities.	HR VPA;VR	Polar Software	161
Ortiz de Guinea, Titah, & Léger (2014)	JMIS	A study to test the effect of factors for PU and PEOU.	Impedance and baseline; artifact correction.	EEG	B-Alert	24
Riedl et al. (2010b)	MISQ	A study on gender difference in online trust.	fMRI data collection; artifact correction.	fMRI	SPM5	20
Riedl et al. (2014b)	JMIS	A trust game to investigate responses to human & avatar faces.	To accommodate subjects to fMRI; fMRI data collection and artifact correction.	fMRI	Presentation/ FSL	18
Tams, Hill, Ortiz de Guinea, Thatcher, & Grover (2014)	JAIS	Multi-method validation between physiological and psychological measures of technostress.	Acquiring pre-task measurements, noise control of sAA data, variables control.	sAA	Salimetrics	64
Teubner et al. (2015)	JAIS	Arousal and bidding behavior with humans/computer opponents.	Signal calibration; Artifacts avoidance.	HR SC	z-Tree/ Ledalab	103
Vance et al. (2014)*	JAIS	An experiment to examine EEG measures for risk perceptions.	Remove artifacts in the EEG data; examine the channels.	EEG	Geodesic EEG (EGI)	59
Warkentin et al. (2016)	JAIS	Analysis on how fear appeals affect user intention to enact secure IT behaviors.	fMRI data collection; artifact correction.	fMRI	E-Prime/ FSL	17

Note: N: total number of participants; *JAIS: Journal of the Association for Information Systems; JMIS: Journal of Management Information Systems; MISQ: MIS Quarterly*; EEG: electroencephalography; EMG: Facial electromyography; ERP: event-related brain Potentials; fMRI: functional magnetic resonance imaging; HR: heart rate; sAA: salivary alpha-amylase; SC: skin conductance; VPA: verbal protocol-analysis; VR: video recording; FSL: fMRI software library; SPM: statistical parametric mapping; z-Tree: Zurich Toolbox for Readymade Economic Experiments; PEOU: perceived ease of use; PU: perceived usefulness; ICT: information & communication technology.

Table 1. Mapping of NeuroIS Experiments to Requirements (Mapping)

Authors (year)	R1a	R1b	R2a	R2b	R2c	R3a	R3b	R3c	R3d	R3e	R4a	R4b	R4c	R4d
Astor et al. (2013)	●	○	●	●	●	○	●	●	●	●	●	●	●	●
Cyr et al. (2009)	●	○	●	●	○	●	●	○	●	●	○	●	●	●
Dimoka (2010)	●	○	◉	◉	○	●	●	●	●	○	●	●	●	●
Fadel et al. (2015)*	●	○	●	●	○	●	●	○	●	○	○	●	●	●
Gregor et al. (2014)	●	○	●	●	○	●	●	●	○	●	●	●	○	●
Hu et al. (2015)	●	○	●	●	○	○	●	○	●	●	○	●	●	●
Kuan et al. (2014)	●	○	●	●	○	●	●	●	●	●	○	●	●	●
Léger et al. (2014)	●	○	●	●	◉	●	●	●	●	●	○	●	●	●
Li et al. (2014)	●	○	●	●	○	●	●	●	●	●	●	●	●	●
Minas et al. (2014)	○	●	●	●	○	●	●	○	●	●	○	●	●	●
Ortiz de Guinea & Webster (2013)	●	○	●	●	○	○	●	●	○	○	○	●	●	●
Ortiz de Guinea et al. (2014)	●	○	●	●	○	●	●	○	○	●	○	●	●	●
Riedl et al. (2010b)	●	○	◉	◉	○	●	●	●	●	●	○	●	●	●
Riedl et al. (2014b)	○	●	◉	◉	○	○	●	○	●	●	○	●	●	●
Tams et al. (2014)	●	○	○	○	○	○	○	○	●	●	○	○	●	●
Teubner et al. (2015)	●	●	●	●	○	○	●	○	○	○	○	●	●	●
Vance et al. (2014)*	●	○	●	●	○	●	●	●	●	●	○	●	●	●
Warkentin et al. (2016)	●	○	◉	◉	○	○	●	●	○	●	○	○	●	●

Note: ○: requirements not needed by experiment; ●: requirements partially needed by experiment and demonstrated in earlier experiments by Brownie; ●: requirements needed by experiment and demonstrated in earlier experiments by Brownie; ◉: requirements needed by experiment, extensible in Brownie, with language wrappers.

Requirement list: R1a: individual interaction; R1b: group interaction; R2a: store biosignal data and event synchronization; R2b: perform signal quality checks; R2c: real-time biosignal processing; R3a: websites; R3b: flexibility of user activity logging; R3c: commonly used programming language; R3d: extensibility; R3e: multimedia; R4a: open source; R4b: tutorials, support; R4c: replicability; R4d: questionnaires.

*: assessment of requirements not validated by author team.

Table 2. Case Study Statistics

Interviewee	Number of formal interactions	Number of informal interactions	Involvement in research phase							
			1	2	3a	3b	4a	4b	5	6
Chief investigator 1 (CI1)	4	2	x	x	x	x	x	x	x	x
Chief investigator 2 (CI2)	2	2	x	x	x	x		x	x	x
Research assistant (RA1)	4	12			x	x	x			
Research assistant (RA2)		2			x	x				
Experiment participant 1 (EP1)	1						x			
Experiment participant 2 (EP2)	1						x			

Note: formal interactions included structured and semi-structured interviews, participation in test sessions in the lab, feedback provided on a written evaluation form, and statements in the experimenters' scientific publication. Informal interactions included e-mail exchanges and telephone conversations.

4.2.2 Phases 1 and 2: Identify Research Questions and Build the Theoretical Foundation

In the first phase, the CIs identified the research questions for the project. First, they planned to investigate whether the crowdfunding mechanism would influence participants' behavior (RQ1). Second, they planned to examine whether interacting with a human or a computer group would influence participants' behavior and perceptions (RQ2). Third, they planned to examine the influence of the crowdfunding mechanisms on the intensity of affective processes (RQ3)[6]. The CIs considered conducting an online study but discarded this option because "given the specific subject of crowdfunding" and the novelty of the approach, they felt they needed the level of control provided by a lab experiment with the appropriate choice of treatments. They were to supplement the experiment with questionnaires to gather additional data required to investigate the above questions. In this phase, the CIs could already clearly perceive requirements R1a, R1b, R3c, and R4d. CI2 explained that:

> *We had to come up with a proprietary user interface, to simulate a crowdfunding setting, which was able to incorporate a number of projects. We wanted participants to see a progress bar (a graphical scale), and to be able to enter information. In addition, we needed to track each and every participant move (or click). We wanted a tool that enabled us to manage all participants to see the same screen and information during a given lab session, as well as a software that allowed us to track the timestamps of user activity. In addition, recording physiological data in the form of heart rate, was necessary, to assess emotional processes without direct reports of participants.*

In summary, participants needed to interact with the interface (R1a) and be grouped differently (R1b), their every click had to be logged (R3c), and a questionnaire had to be included (R4d). The CIs realized they needed to gather physiological data in order to assess affective processes during participants' decision making (R2a) towards the end of the conceptual phase. CI1 stated in the interview that one important issue was extending the round-based experiment to one in which participants could dynamically view others' actions (i.e., from a static experiment to one with a high level of real-time interaction). Hence, the CIs stated extensibility (R3d) as an important criterion for experimental software choice as well.

4.2.3 Phase 3a: Design of the Experiment

The CIs decided on a between-subject design with different interfaces for two treatments (two crowdfunding mechanisms). With respect to group interaction (R1b), they assigned 12 participants to each session that, in turn, comprised 24 periods. In each period, the CIs randomly assigned participants to two groups of six with the restriction that two participants would never be assigned to the same group twice. The design included three phases or screens (information and input, round result, final result) with interaction elements such as standard input boxes and buttons and graphical elements such as status bars, a timer (ticking down from 60 seconds), and result tables. After each user input phase, an arrow or a cross would indicate which projects were funded. Hence, a common programming language that offered

[6] The third question emerged at a later point in time than the first two questions.

libraries for these elements and supported their future planned extensions emerged as an auxiliary requirement of the experiment (R3c) at this stage.

The CIs then turned to the question of which experimental software would fulfill their requirements best. LimeSurvey[7] offered the required questionnaire functionality but did not offer biosignal acquisition or processing nor sufficient flexibility in interface and experimental flow design. RA1 explained in their written feedback that, "[The] crowdfunding setting required a system that allowed us to specify detailed endowment rules and an advanced client-server communication". They considered z-Tree as another option but, as CI1 pointed out in the interview, "choosing [z-Tree] would not have enabled us to extend the experiment to a real-time setting [R3d]". CI2 stated in the interview that using z-Tree would have required the CIs to drop R2 since one could not acquire biosignal data in z-Tree at the time[8]. With regard to interface design (R3c), CI1 considered both the z-Tree toolbox and other candidate software to be "limited in possibilities". Finally, CI1 and CI2 chose Brownie as their experimental platform since it fulfilled the primary requirements of the experiment (R1a, R1b, R2a, R2b, R3d, R4c, and R4d). CI1 and CI2 stated that system usefulness (i.e., that the system would meet all their requirements and be able to cope with their "special wishes") was initially the most important criterion in choosing Brownie.

4.2.4 Phase 3b: Implementation of the Experiment

In Phase 3b, RA1 and RA2 implemented the design specifications. The CIs provided PowerPoint slides for the screen designs, Excel files for the group-matching logic, screenshots of whiteboard discussions, mock-ups, and textual descriptions. RA1 and RA2 were also involved in the CIs' discussions. RA2 had basic programming skills and prior experience in Java from university courses. RA1, having participated in programming projects before, had advanced programming skills. Both CIs estimated that, from their experience, it took them a normal amount of time (two weeks' time) to find a student research assistant reasonably fluent in Java (i.e., no longer than it would have taken to find a person with command of another programming language). CI1 observed that, in the beginning of the project, it was difficult to estimate the level of programming skills required for the project. Due to the complexity of the experiment and the overhead of learning to work with a new experimental software, the CIs expected that they would run into difficulties implementing the experiment. CI1 explained that:

> *It would have been easier to find someone to program for z-Tree, especially for simpler experiments, but not for programming our experiment, which would have been a bigger challenge. z-Tree would have been useful for programming simpler settings, such as a standard public goods game, but for the crowdfunding setting we wanted to examine, along with physiological measures, Brownie turned out to be the better choice. In addition, on z-Tree, it is not possible to use common knowledge about programming, which makes one heavily dependent on the manual, whereas on Brownie, we could find help for new ideas online, for instance on forums, very easily.*

Based on feedback from CI1 and CI2, RA1 altered the implementation iteratively. RA2 worked 10 hours per week on the implementation and 80 hours in total, which included the setup and learning time. The implementation phase comprised the following four major steps:

- **S1: Installation and setup (11 March to 13 March, 2015):** RA1 and RA2 requested access to Brownie's source code via email. We sent them the source code with links to tutorials, sample experiments, and video tutorials. RA1 found the information easy to understand and effective for installing Brownie, which points to a good information quality. CI1 remarked that "it would be helpful to have more of these great tutorials, but it is a good start". CI2 noted that it took more work than they had expected to install and setup Brownie, to understand the various components, and the interaction between them. However, as soon as they gained a basic understanding, they could make progress much more easily. CI1 recalls that RA1 was "frustrated in the beginning, but then with the video and online tutorials, they were able to set it up fast, and get it working". CI1 also noted that they would expect gaining a basic understanding to be a problem with any new experimental tool.

[7] LimeSurvey is a popular open source Web application to conduct surveys. It is available online on http://limesurvey.org.
[8] We note that there are in fact NeuroIS experiments that use z-Tree. However, prior experiments and the authors of z-Tree state that one cannot integrate biosignals directly with z-Tree and that it has limited logging functionality (refer to Teubner et al.'s (2015) experiment in Table 1 in this paper).

S2: UI and client screen design (16 March to 15 April, 2015): based on the conceptual designs from phase 3a, RA1 first designed the interfaces participants would interact with, which took approximately 40 hours. During this period, RA1 asked questions about quickly implementing the GUI and the look-and-feel of client screens, which the developer team answered by providing instructions for Window Builder and improving the tutorials on how to implement the GUI with Window Builder. With respect to the interface quality, RA1 stated in written feedback evaluation that: “After finding out how to do it, it is neat to be able to design an attractive experiment (using the WindowBuilder for eclipse) with Java”.

S3: Experiment client-server communication (23 April to 28 April, 2015): RA1 proceeded with implementing the experimental flow (i.e., the ordering of displaying screens, sending client-specific information from the server, and incorporating round-based, client-level, and group-level logic in the experiment). The developer team assisted in this step by pointing to current experiments and tutorials. RA1 stated in the written feedback that: “It was rather easy to include and start the experiment with its server/client interfaces”. CI2 found that “incorporating message passing client-server interaction was relatively easy,” and CI1 did not “remember any problems in this step”. The CIs and RAs considered the tutorials as particularly helpful for solving issues, which points to Brownie’s fulfilling R4b. CI2 commented on the information quality of the support material: “The information (such as videos, documentation) provided with Brownie as well as online support in forums was really helpful for finding the information required to complete the experiment”. CI1 recommended for future work on Brownie that “the website’s search and overview functionality could be improved such that tutorials can be found faster”. CI1 emphasized the importance of R1a and R1b for their study design in written feedback: “The interaction of groups across different cohorts was really important to us. Specifically, group contributions and funding had to be calculated for each of the four projects and displayed to all participants. Brownie enabled us to achieve this”. In order to ensure that participants were not placed in the same group twice, the subject group allocation for each round had to be fixed in the implementation while ensuring balanced treatments. The existing features of Brownie at that point in time (random vs. factorial) did not include this function. The developer and experimenter teams discussed this issue, and the developer team decided to extend Brownie to incorporate customized matching. It is now available for all experiments. This special requirement demonstrates R3d (i.e., the system’s extensibility for a specific need). In their written feedback, RA1 commented on system usefulness: “It was nice to have pretty much all the freedom and possibilities anyone could ever ask for concerning the backend and which (matching) algorithms you want to include”.

S4: Testing (18 May to 11 June, 2015): The CIs and RAs completed this step in several iterations throughout the implementation process both in offline software testing and session testing in the lab. CI1 stated that it took some effort to start multiple clients (12 in this case) on the same system and test the experiment, but they managed to complete it successfully[9]. CI2 stated that, initially, setting up each computer in the lab separately was a bit inconvenient. However, they later implemented a script based on a previous example for another experiment, which solved this issue and made lab testing very easy. The solving of this issue demonstrates the advantage of using a common programming language (R3c) and the platform’s extensibility to easily incorporate desirable standard features (R3d). Testing took approximately a quarter of the overall development time of the research assistant (approximately five work days). CI2 considered the tutorials, especially videos, to be “very helpful”. CI2 would rate the difficulty of learning to use Brownie as “intermediate” (based on easy, intermediate, and hard as the rating scale). CI1 found it more difficult to get a basic understanding of Brownie (see Section 1) but thought that, thereafter, it was quite easy to understand how to adapt Brownie to different purposes and current and future research designs. In total, with respect to the overall satisfaction, all team members stated a satisfaction level of 4 on a scale of 1 to 5. They said that difficulties in implementation partly resulted from the complexity of the experiment design and partly from insufficient documentation at the beginning of their research project. Since then, [the developers of Brownie have extended and elaborated on support materials and tutorials: tutorials for all experiments are now publicly available.

[9] At this point, the developers created a solution that enabled one to automatically start and connect multiple clients, which they integrated into the experiment and is now available for all experiments.

4.2.5 Phase 4a: Conduct the Experiment and Collect Data

The CIs conducted the study with 12 participants per session and one session for each treatment (24 participants in total). CI1 stated that, with respect to interface quality, they could:

> *Successfully and comfortably manage the flow of the experiment (such as configuring sessions, viewing session and client statuses during run-time), as well as programming additional elements (such as questionnaires), easily in the experiment using Brownie.*

CI2 added that the “front end was really good, and we needed minimal help in using it to manage the experiment. There were no issues during run-time of the session.”. CI1 stated that they incorporated pre-experiment questionnaires without any difficulty. The CIs collected post-experiment questionnaires via a separate Google forms website, which they invoked from within Brownie as part of the website integration functionality (R3a). The advantage in invoking the Google from within Brownie was that, since the experiment was running in full-screen mode, participants could not wrongly click on any other parts of the window or close the form accidentally.

In addition to behavioral data, the experimenters also collected ECG and EDA data using the Bioplux (2007) sensor system, which was transmitted using Bluetooth and stored on the individual participants’ PCs, which demonstrates Brownie’s fulfilling R2a. CI2 stated that “Brownie gave us the opportunity to log any user event in the experiment and synchronize them with the physiological data with the required timestamp information”, which demonstrates Brownie’s fulfilling R2b. They could complete the experiment successfully and integrate data for their analysis, which demonstrates Brownie’s fulfilling R3a (integration of websites) and R4d (implementation of questionnaires). CI1 emphasized that the flexibility aspect (incorporation of new ideas) was the most satisfying aspect of programming with Brownie, and CI1 felt “limitless” in terms of what they would be able to implement in their project. RA1 summarized in the written feedback that: “All in all, Brownie worked as we wanted it to”. CI1 added that Brownie matched their expectations with respect to system usefulness and overall satisfaction.

In order to gauge the participants’ views in using the interfaces implemented in Brownie, we interviewed two experiment participants (EP1 and EP2) to obtain qualitative feedback on the platform’s usability. EP1 had already taken part in experiments on z-Tree and Brownie and had taken part in similar group interaction scenarios as well. EP1 stated that the experiment interface was self-explanatory and easy to interact with due to its clear structure. EP1 felt absorbed in the experiment but would have liked to avoid “uninteresting” waiting times while other participants were finishing their respective tasks. These waiting times resulted from the CIs’ decision to go with a round-based design; they indicated in the interview that they planned on changing to a dynamic design in a follow-up experiment. EP1 did not discern any visible differences between using interfaces programmed with Brownie or other experimental software. The participant suggested that incorporating a chat function would have enabled communication with other participants. Although a client-to-client communication feature is available in Brownie, the CIs had decided against enabling it in order to maintain a higher level of experimental control. Comparing two experimental interfaces implemented in Brownie (both of which the participant had taken part in), EP1 rated the satisfaction level of the crowdfunding experiment as 4 on a scale of 1 to 5 and the interface of the other experiment as 5. That interface had contained real-time elements (such as live charts and trends, which Brownie also features), which appealed more to the participant. EP1 had taken part in another experiment with physiological measurements before and found the overall experimental interface to be engaging.

EP2 had not taken part in an experiment with Brownie before but had in several experiments with z-Tree. They found participating in the crowdfunding experiment interesting and thought it felt real to an extent. EP2 felt that this was one of the “better” experiments they had participated in so far, that the interface was absorbing and fun, and that they were not bored during the experiment. Comparing Brownie with z-Tree, EP2 stated that the structure of the crowdfunding experiment was comparable (information, decision, and result).They observed that, compared to previous experiments on z-Tree, the loading time of experiment screens, especially those with a lot of information, was much shorter. In the previous experiments, they had found the long loading times of such screens rather tedious. On a scale of 1 to 5, EP2 rated the satisfaction level of the crowdfunding experiment as 5. In summary, the above statements point to a satisfactory and engaging interface quality of the participant screens.

4.2.6 Phase 4b: Analyze Data

With respect to flexibility of user data logging (R3b) and system usefulness, RA1 stated in written feedback that: "[Brownie was] used for data storage, and data storage methods used in previous experiments were useful examples to easily integrate these in our experiment" and also that "the pre-defined database helped a lot". This database refers to Brownie's underlying database schema, which we designed to allow one to easily query and reorganize experimental data. In particular, the CIs stored client data by concatenating relevant user values (e.g. input and click data) along with the respective client timestamps in the central server database. To retrieve more information on different granularities such as subjects, round, the CIs used group-level join queries, which increased the system's usefulness in terms of data storage's flexibility. CI1 said that:

> *[Working with] normal [i.e., observed data about participant behavior] data was no problem. Brownie further enabled the synchronization of physiological data, with the events of the experiment, by adding timestamp entries. These time-synced event entries were later useful for analysis of physiological data. However, physiological data extraction can be improved.*

CI1 specifically suggested reducing the steps necessary for transforming and analyzing the stored raw physiological values.

4.2.7 Phases 5 and 6: Interpret the Experimental Data and Discuss the Results

The CIs proceeded to analyze the experimental data and publish their findings in suitable outlets. In future sessions, they planned to store real-time processed physiological data in addition to the raw data. CI1 observed that:

> *In all our decisions, we never came to a point where we could not do something because it was not possible on the platform. There was always a solution, and this was a very positive aspect for us." CI2 gave as their opinion that, "[Brownie] is an excellent alternative to z-Tree. The possibility to include physiological measurements is really important, since they help understand participant behavior greatly.*

CI1 considered physiological data to be "honest" in terms of showing whether participants were aroused, annoyed, or bored during an experiment. Both CIs opined that Brownie's integrating physiological measures would be extremely useful for upcoming studies in the IS domain. If CI2 were to do another experiment, they would "implement it with Brownie". CI1 and CI2 stated that they did not notice a marked difference to other experimental software while conducting their experiment since the core building blocks (e.g., session, treatment, and periods) were similar to other software they used before. In future research, they planned to extend their experiment by new treatment levels, which demonstrates Brownie's fulfilling R4c (i.e. the replicability of an experiment that uses Brownie) and also shows the overall satisfaction with Brownie, which CI1's statement that "I would be comfortable to design and create my next planned experiments in Brownie, and I would recommend it to other researchers" emphasizes. The CIs also stated that, with the benefit of hindsight, the quality of the available information played a large role in helping them to successfully complete their project.

4.3 Evaluation of Usability Based on an Experimental Study

4.3.1 Experimental Study Description

We evaluated the usability of Brownie for experimenters by conducting a laboratory study to compare Brownie to the current gold standard in software for behavioral experiments, z-Tree. The study task comprised altering a provided ultimatum game to a trust game. Specifically, participants had to 1) change the experimental logic, 2), change the interface, 3), add player pictures, 4) display a reputation score, and 5) include biosensor measurements. The ultimatum game is a two-person game in which the first player (the requester) receives a sum of money and proposes how to divide the sum between himself and the other player. The trust game differs from the ultimatum game in that the money proposed by the requester is tripled and forwarded to the responder, and the responder has a choice about how much of the now-tripled money to send back to the requester. We chose this setting since research has used the ultimatum game (Joe et al., 2008) and the trust game to understand peer-to-peer interactions and trusting behavior in online environments (Du, Huang, & Li, 2013; Riedl et al., 2014b; Ananthakrishnan, Li, & Smith, 2015).

4.3.2 Procedure

We performed the study with a within-subject design, where a participant would implement the experiment in both Brownie and z-Tree. Both treatment sessions (Brownie and z-Tree) took place twice in two subsequent weeks at the same time of day and the same laboratory. We randomly assigned participants to one of the treatments in the first week and alternated participants to the other treatment in the second week.

The study proceeded as follows. First, we gave participants general instructions on the study task. Second, we had them fill in a pre-study questionnaire on their programming abilities. Third, we gave them a tutorial explaining the ultimatum and trust game rules, how the ultimatum game was implemented in the software, and which changes were necessary to alter it into a trust game. In order to ensure that the depth of the instructions and the information on the tutorials for both software were comparable, we adapted the same tutorial for the other and made modifications only with respect to the software-specific terms while retaining the task descriptions. We divided the study task into eight subtasks. After each subtask, we gave participants a short questionnaire on how difficult they felt the task had been. After the last task, we gave participants a final questionnaire that asked for their overall usability evaluation of the software. We measured usability with the IBM usability scale (Lewis, 1995). Participants had a time limit of 120 minutes. Prior to the study, we estimated the time required for each subtask by timing six student assistants with no or limited prior experience with Brownie and z-Tree and averaging their results.

4.3.3 Measures

We assigned task scores based on successful lines of code that were functionally correct for performing the given task. Hence, we broke down each task down into three or four code alterations, which we specified for both software. Two independent researchers scored the degree of task completion based on the solution code submitted by participants. To measure usability, we used three factors from the IBM usability scale: system usefulness, information quality, and interface quality. Confirmatory factor analysis with R-3.3.2 and lavaan 0.5-20 showed good model fit (Appendix C).

4.3.4 Sample

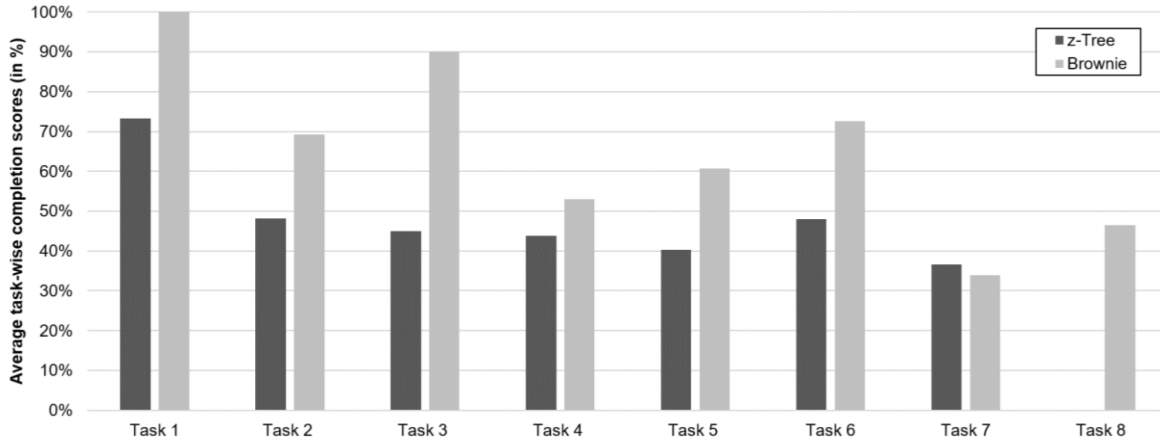
We invited students from a course on experimental methods in business research to ensure that all participants had a basic understanding of and a general interest in experiments and experimental research. In total, 28 students participated. For their participation in the usability study, we awarded students with course credit points. We carried out the study in the experimental lab at Karlsruhe Decision and Design Laboratory (KD2Lab). Prior to our study, the ultimatum game had been briefly explained within the scope of the course, but the trust game had not. Experimental software had not been discussed in the course up to this point. As for the participants, 25 percent of participants were female, and they were all pursuing a master's degree. Average session duration was 1:33 hours in Brownie, and 1:26 hours in z-Tree.

4.3.5 Results

Figure 2 shows the average task scores of our participants for z-Tree and Brownie, which indicates how well they solved each task. Task 8, the configuration of biosensors, was unique to Brownie. Our results indicate that, on average, participants performed better on Brownie with the exception of task 7 (accessing experimental data). Taking the results by session, participants performed better in both software pieces (Brownie and z-Tree) in the second session (completion rates increased from the first session by 19% in Brownie and 10% in z-Tree), which one can explain by their already having learnt about the study task.

Our results show that Brownie compares well to z-Tree; participants rated it as good as z-Tree in both overall usability and all three usability subscales (Figure 3). Mean and median ratings of all scales were lower for both software pieces (Brownie and z-Tree) in the second session but not significantly so.

In an open feedback question at the end of the second sessions, we asked participants to indicate whether they would prefer to use Brownie or z-Tree in the future. Twelve participants indicated a preference for Brownie and cited reasons such as flexibility, common programming language, open source, and more intuitive programming. Eight participants preferred z-Tree because they found it easier to understand. Six participants did not express a preference, and two participants explicitly noted that they were unable to decide: they stated that they found Brownie more flexible and powerful but z-Tree easier to get started with.



Note: Task 1: Running provided ultimatum game; Task 2: Modifying logic to trust game; Task 3: Modify screen UI with 3 new fields; Task 4: Adding action logic to button; Task 5: Modifying server to trust game logic, and log user data in database; Task 6: Modify screen by adding an Image and a reputation score; Task 7: Access database and view data; Task 8: Configure sensors and acquire sensor data.

Figure 2. Task Completion Rates for Brownie and z-Tree

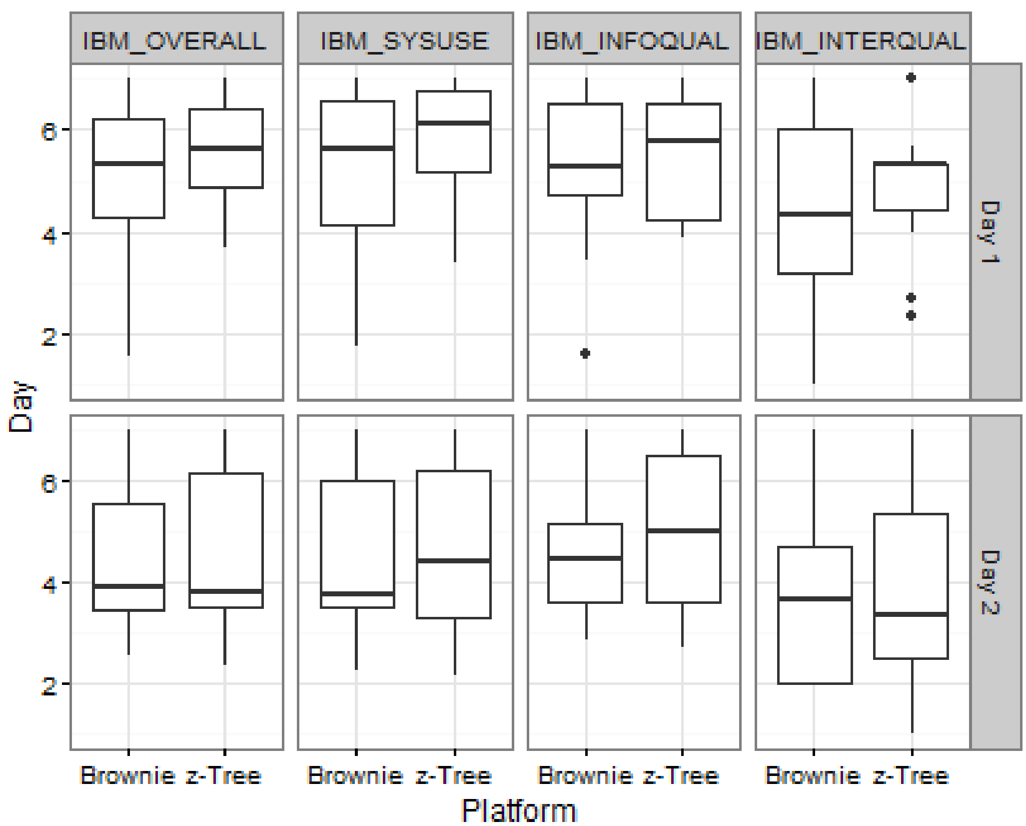


Figure 3. Usability Scores for Brownie and z-Tree[10]

[10] Note: the y-axis indicates usability ratings aggregated across subjects, software, and on two different days on a scale of 1-7. Smaller box plots indicate higher agreement within subjects, which is observed in the case of z-Tree in Day 1, and Brownie for Day 2. Differences between the groups were not visible, denoting which denotes comparable usability. Long lower (upper) whiskers indicate that participants varied more over the lower (upper) quartile of the usability score.

Taken as a whole, the results of the experiment show that, in terms of usability (while considering common aspects of experiments such as experiment logic and UI modification), Brownie compares well with z-Tree. At the same time, Brownie offers additional NeuroIS-specific features that z-Tree does not include (R2a, R2b, R2c) and possibilities for further extension (R3a, R3c, R3d, R4a). Given that both our literature review and case study confirm the requirement list we identify earlier in the paper, we conclude that Brownie meets the emerging requirements of NeuroIS research and cognate areas, which current experimental software does not currently fulfill.

5 Discussion and Future Work

We designed, implemented, and evaluated the experimental platform Brownie to facilitate NeuroIS research in the lab. We confirmed the platform's usefulness and usability in three evaluations: a literature review of current NeuroIS studies to verify its usefulness, a case study of a research project implemented using Brownie to evaluate its usefulness and usability in a complex setting, and an experimental study to evaluate its usability in comparison to an alternative experimental software. The results of our evaluation show where Brownie's advantages (compared to alternative experimental software) lie and how issues faced while implementing experiments can be solved with Brownie.

5.1 Contribution and Communication

Brownie is a stable platform for implementing individual and group interaction experiments that integrate neurophysiological measurements. In terms of the knowledge contributions of design science research (Gregor & Hevner, 2013), one may categorize Brownie as an invention in that it applies a new solution (synchronous biosignal acquisition and real-time processing) to a new problem (behavioral research using biosignals). Brownie is an invention in the sense that it is an artifact that one can apply and evaluate in a real-world context, and a key contribution is our conceptualizing the problem itself (i.e., lack of extensible and open source software for NeuroIS researchers). In addition, Brownie differs from routine design, improvements, and exaptation by having a particularly low application domain maturity and solution maturity according to Gregor and Hevner's (2013) framework.

With the development and distribution of Brownie, we contribute to interdisciplinary IS research by reducing obstacles (especially related to IT infrastructure and IT expertise) to conducting experimental research on NeuroIS. Cognate areas such as behavioral economics, experimental psychology, and affective computing can profit from Brownie due to its extensive capabilities for implementing complex human-computer and group interaction scenarios. Brownie reduces barriers for NeuroIS researchers by providing a platform to collaboratively develop and test new system features (NeuroIS and non-NeuroIS) that researchers can use across experiments. For instance, due to its extensibility and open source distribution, Brownie is innovative in that it can support researchers to develop and agree on methodological guidelines by implementing reference solutions and providing them to the research community. Moreover, by distributing standard experimental setups (e.g., standard scenarios such as the Trust Game) along with it, Brownie helps researchers replicate prior studies (e.g., to test effect stability across samples) and implement extensions and additional treatments for their experiments.

In terms of dissemination and communication, we discussed the need for an experimental platform at several workshops with researchers from the NeuroIS domain. We distribute the source code for the platform as a ready-to-use eclipse workspace project with version controlling implicitly enforced through the use of the well-known distribution service Bitbucket. Along with the source code, we distribute a wiki with a setup tutorial, a quick start guide, and a step-by-step guide to programming and managing an experiment[11]. In addition, we make available video tutorials on "how to set up Brownie", and "how to set up an experiment" at https://www.youtube.com/channel/UCIwooE5L0D0FTGi1RtIuPJg. We have also prepared more tutorials that illustrate the use cases in Appendix B. Finally, we communicate the iterations of Brownie's design and development via structured workshops.

Application areas for Brownie in interdisciplinary IS research include the design of websites and decision support system interfaces (e.g., to study the variability of user attention and differences in information search behavior or to determine optimal information levels for users based on cognitive workload) (Kohavi, Henne, & Sommerfield, 2007; Léger et al., 2014; Seuken et al., 2012). Further research areas that Brownie suits include the influence of emotions and social context on individual decision making. For

[11] https://bitbucket.org/kit-iism/experimenttool/wiki/Home

instance, one can use Brownie to examine how trust is established in e-commerce contexts (Gefen, 2002; Riedl et al., 2010b; Du et al., 2013; Adam, Krämer, & Müller, 2015) and how it affects consumer choice and purchase decisions (Dorner et al., 2013). Moreover, one can use Brownie to develop, test, and evaluate neuro-adaptive systems via its real-time signal processing capabilities. Finally, one can also use Brownie to simulate complex interactions of users with computer agents and purely agent-based interactions where required.

5.2 Limitations and Future Work

Brownie at its current form has limitations, some of which future research could address by extending the platform. First, in its current form, Brownie does not allow one to directly integrate neuroimaging techniques (such as fMRIs or functional near infra-red spectroscopy (fNIRS)). Such imaging techniques are vital in determining the brain activity of the above-mentioned IS constructs. For software technology, researchers of these technologies predominantly use programs based on Python, Matlab, or C++ to acquire data and specialized Matlab-based libraries to analyze such data. One can extend Brownie to communicate with other programs by providing suitable wrappers to other languages to facilitate communication with a broader range of sensors. Examples of such wrappers include Jython (a Java Python interpreter) and MatlabControl[12] (a program used to call Matlab functions from within Java).

Second, the paper conceptualizes the design science problem via the proposed requirement set. Although we present a design artifact, evaluate it, and provide directions for further development, one may also validate the requirement set via other design instantiations (Gregor & Hevner, 2013, p. 10). These instantiations could adopt a different design artifact for each requirement, such as choosing a different language for the implementation. Other design instantiations may arise as a result of further experiments conducted with Brownie or by replicating existing experiments in different geographical or cultural contexts.

Third, the real-time signal processing capabilities in Brownie are based on heart rate and skin conductance sensors. One ought to enhance the processors used to compute the respective features in terms of efficiency and real-time aspects and in terms of extensibility to other sensors, such as PPG, EMG, and EEG (Riedl et al., 2014a; Müller-Putz et al., 2015). The modular architecture of Brownie facilitates extensions of the real-time features, such as creating different manifestations of live-biofeedback, altering and adapting the interface based on neuro-information, and so on. Most importantly, integrating further real-time features in Brownie will impose different challenges for different biosignals. For instance, processing EEG data in real time, such as in the domain of brain-computer interfacing, requires substantial efforts in terms of artifact detection (Müller-Putz, Riedl, & Wriessnegger, 2015). In addition, integrating statistical and data analysis modules, such as R-packages, in Brownie would make it possible to analyze experimental data in the platform if required. These analysis modules could enable one to build individual variances in real-time sensor-data processing based on participants' gender, age, or historical data of.

Finally, with the proliferation of mobile devices and the variety of tasks being performed on them, researchers need to conduct NeuroIS experiments across different devices to understand user engagement with mobile devices (Bhandari, Neben, & Chang, 2015; Hollingsworth & Randolph, 2015). Brownie currently does not support experiments with mobile applications, and, in the future, one could extend it to incorporate experiments applying NeuroIS methods with mobile applications and sensors.

6 Conclusion

Brownie serves as a solution that meets emerging requirements for conducting NeuroIS research. At present, the effort and technical knowledge required to conduct NeuroIS research are substantial. With Brownie, we help to reduce both the effort and the technical knowledge demands to a better manageable level. Brownie is innovative in that it reduces barriers for IS researchers to engage in collaborative research and has a high degree of flexibility with regards to individual experimenters' needs, such as extensibility to emerging NeuroIS methodologies and scalability across various devices. We hope that Brownie contributes to future research in individual and group interactions by leveraging existing and emerging NeuroIS methods for user interaction research and that it fosters collaborative and interdisciplinary research across domains by facilitating replicability and the exchange of research knowledge.

[12] http://www.cs.virginia.edu/~whitehouse/matlab/JavaMatlab.html

Acknowledgments

We thank the contributions of the following students who contributed to several aspects of the platform: Martin Caspari, Tonda Roder, Philipp Rouast, Bahubali Angol, and Dominik Jung. We also thank participants of the Gmunden Retreat on NeuroIS, Gesellschaft fuer Konsumforschung, and the research team of Daniel Schnurr and Niklas Horstmann for their valuable comments and feedback. We also thank Claudia Niemeyer and her research team for their support in conducting the case study and Florian Hawlitschek, for his implementation ideas for the platform and for his support in conducting the usability study.

References

Adam, M. T. P., Krämer, J., & Müller, M. B. (2015). Auction fever! How time pressure and social competition affect bidders' arousal and bids in retail auctions. *Journal of Retailing, 91*(3), 468-385.

Adam, M. T. P., Krämer, J., Jähnig, C., Seifert, S., & Weinhardt, C. (2011). Understanding auction fever: A framework for emotional bidding. *Electronic Markets, 21*(3), 197-207.

Ananthakrishnan, U., Li, B., & Smith, M. (2015). A tangled web: Evaluating the impact of displaying fraudulent reviews. In *Proceedings of the International Conference on Information Systems.*

Astor, P. J., Adam, M. T. P., Jerčić, P., Schaaff, K., & Weinhardt, C. (2013). Integrating biosignals into information systems: A NeuroIS tool for improving emotion regulation. *Journal of Management Information Systems, 30*(3), 247-278.

Benbasat, I., Dimoka, A., Pavlou, P. A., & Qiu, L. (2010). Incorporating social presence in the design of the anthropomorphic interface of recommendation agents: Insights from an fMRI study. In *Proceedings of the International Conference on Information Systems.*

Benedek, M., & Kaernbach, C. (2010). A continuous measure of phasic electrodermal activity. *Journal of Neuroscience Methods, 190*(1), 80-91.

Bhandari, U., Neben, T., & Chang, K. T. (2015). Mobile app preferences: What role does aesthetics and emotions play? In F. Davis, R. Riedl, J. vom Brocke, P. M. Léger, & A. Randolph (Eds.), *Information systems and neuroscience* (LNISO, vol. 10, pp. 161-165). Berlin: Springer.

Bioplux. (2007). *Wearable body sensing platform* (computer software). Lisbon, Portugal: PLUX Wireless Biosignals. Retrieved from http://www.plux.info

Bostian, A. J. A., & Holt, C. A. (2013). Veconlab classroom clicker games: The wisdom of crowds and the winner's curse. *The Journal of Economic Education, 44*(3), 217-229.

Brainard, D. H. (1997). The psychophysics toolbox. *Spatial Vision, 10*(4), 433-436.

Charland, P., Léger, P.-M., Sénécal, S., Courtemanche, F., Mercier, J., Skelling, Y., & Labonte-Lemoyne, E. (2015). Assessing the multiple dimensions of engagement to characterize learning: A neurophysiological perspective. *Journal of Visualized Experiments*, *101*, e52627.

Chilton, L. B., Sims, C. T., Goldman, M., Little, G., & Miller, R. C. (2009). Seaweed: A Web application for designing economic games. In *Proceedings of the ACM SIGKDD workshop on Human Computation* (pp. 34-35).

Cox, J. & Swarthout, J. (2005). EconPort: Creating and maintaining a knowledge commons (No. 2006-06). *EconPapers.* Retrieved from http://EconPapers.repec.org/RePEc:exc:wpaper:2006-06

Cyr, D., Head, M., Larios, H., & Pan, B. (2009). Exploring human images in website design: A multi-method approach. *MIS Quarterly, 33*(3), 539-566.

Dimoka, A. (2010). What does the brain tell us about trust and distrust? Evidence from a functional neuroimaging study. *MIS Quarterly, 34*(2), 373-396.

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

Dimoka, A., Pavlou, P. A., & Davis, F. D. (2011). NeuroIS: The potential of cognitive neuroscience for information systems research. *Information Systems Research, 22*(4), 687-702.

Djajadiningrat, T., Geurts, L., Munniksma, P. R., Christiaansen, G., & de Bont, J. (2009) Rationalizer: An emotion mirror for online traders. *Proceedings of the 5th International Workshop on Design and Semantics of Form and Movement* (pp. 39-48).

Dorner, V., Ivanova, O., & Scholz, M. (2013). Think twice before you buy! How recommendations affect three-stage purchase decision processes. In *Proceedings of the International Conference on Information Systems.*

Draine, S. (1998). *Inquisit* (computer software). Seattle, WA: Millisecond Software.

Du, N., Huang, H., & Li, L. (2013). Can online trading survive bad-mouthing? An experimental investigation. *Decision Support Systems, 56*(1), 419-426.

Fadel, K. J., Meservy, T. O., & Jensen, M. L. (2015). Exploring knowledge filtering processes in electronic networks of practice. *Journal of Management Information Systems, 31*(4), 158-181.

Fischbacher, U. (2007). z-Tree: Zurich toolbox for ready-made economic experiments. *Experimental Economics, 10*(2), 171-178.

Friedman, D., & Sunder, S. (1994). *Experimental methods: A primer for economists*. Cambridge: Cambridge University Press.

Gefen, D. (2002). Customer loyalty in e-commerce. *Journal of the Association for Information Systems, 3*(1), 27-51.

Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly, 37*(2), 337-355.

Gregor, S., Lin, A. C. H., Gedeon, T., Riaz, A., & Zhu, D. (2014). Neuroscience and a nomological network for the understanding and assessment of emotions in information systems research. *Journal of Management Information Systems, 30*(4), 13-48.

Greiner, B. (2004). An online recruitment system for economic experiments. In K. Kremer & V. Macho (Eds.), *Forschung und wissenschaftliches Rechnen* (pp. 79-93). Göttingen, Germany: Gesellschaft für wissenschaftliche Datenverarbeitung.

Hariharan, A., Adam, M. T. P., & Fuong, K. (2014). Towards understanding the interplay of cognitive demand and arousal in auction bidding. In *Proceedings of the European Conference on Information Systems.*

Haxby, J. V., Parasuraman, R., Lalonde, F., & Abboud, H. (1993). SuperLab: General-purpose Macintosh software for human experimental psychology and psychological testing. *Behavior Research Methods, Instruments, & Computers, 25*(3), 40-–405.

Hendriks, A. (2012). SoPHIE: Software platform for human interaction experiments. Germany: University of Osnabrück.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75-105.

Hollingsworth, C. L., & Randolph, A. B. (2015). Using NeuroIS to better understand activities performed on mobile devices. In *Information Systems and Neuroscience* (pp. 213-219). Berlin: Springer.

Hu, Q., West, R., & Smarandescu, L. (2015). The role of self-control in information security violations: Insights from a cognitive neuroscience perspective. *Journal of Management Information Systems, 31*(4), 6-48.

iMotions. (2016). *Biometric research platform* (computer software). Copenhagen, DK: iMotions A/S. Retrieved from https://imotions.com

Jarvis, B. G. (2003). *DirectRT* (computer software). New York, NY: Empirisoft. Retrieved from http://www.empirisoft.com/directrt.aspx

Jarvis, B. G. (2004). *MediaLab* (computer software). New York, NY: Empirisoft. Retrieved from http://www.empirisoft.com/medialab.aspx

Jerčić, P., Astor, P. J., Adam, M. T. P., Hilborn, O., Schaaff, K., Lindley, C., Sennersten, C., & Eriksson, J. (2012). A serious game using physiological interfaces for emotion regulation training in the context of financial decision-making. In *Proceedings of the European Conference on Information Systems.*

JessX. (2010). *JessX: Java experimental simulated stock exchange* (computer software). Lille, France. Retrieved from http://jessx.ec-lille.fr/

Joe, S. W., & Lin, C. P. (2008). Learning online community citizenship behavior: A socio-cognitive model. *CyberPsychology & Behavior, 11*(3), 367-370.

Kjærgaard, A. L., & Jensen, T. B. (2014). Using cognitive mapping to represent and share users' interpretations of technology. *Communications of the Association for Information Systems, 34*(57), 1097-1114.

Kohavi, R., Henne, R. M., & Sommerfield, D. (2007). Practical guide to controlled experiments on the Web: Listen to your customers not to the hippo. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 959-967).

Kuan, K. K. Y., Zhong, Y., & Chau, P. Y. K. (2014). Informational and normative social influence in group-buying: Evidence from self-reported and EEG data. *Journal of Management Information Systems, 30*(4), 151-178.

Léger, P.-M. (2006). Using a simulation game approach to teach enterprise resource planning concepts. *Journal of Information Systems Education, 17*(4), 441-448.

Léger, P.-M., Sénecal, S., Courtemanche, F., Ortiz de Guinea, A., Titah, R., Fredette, M., & Labonte-LeMoyne, É. (2014). Precision is in the eye of the beholder: Application of eye fixation-related potentials to information systems research. *Journal of the Association for Information Systems, 15*(10), 651-678.

Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction, 7*(1), 57-78.

Li, M., Jiang, Q., Tan, C.-H., & Wei, K.-K. (2014). Enhancing user-game engagement through software gaming elements. *Journal of Management Information Systems, 30*(4), 115-150.

Loewenstein, G. (2001). The creative destruction of decision research. *Journal of Consumer Research, 28*(3), 499-505.

Minas, R. K., Potter, R. F., Dennis, A. R., Bartelt, V., & Bae, S. (2014). Putting on the thinking cap: Using NeuroIS to understand information processing biases in virtual teams. *Journal of Management Information Systems, 30*(4), 49-82.

Moody, G. B., Mark, R. G., Goldberger, A. L. (2001). PhysioNet: A Web-based resource for the study of physiologic signals. *IEEE Engineering in Medicine and Biology, 20*(3), 70-75.

Müller-Putz, G. R., Riedl, R., & Wriessnegger, S. C. (2015). Electroencephalography (EEG) as a research tool in the information systems discipline: Foundations, measurement, and applications. *Communications of the Association for Information Systems, 37*, 911-948.

Neuroscan (2008). *NeuroScan 4.5* (computer Software). Charlotte: NC: Compumedics. Retrieved from http://compumedicsneuroscan.com/

Nogueira, P. A., Aguiar, R., Rodrigues, R., & Oliveira, E. (2014). Designing players' emotional reaction models: A generic method towards adaptive affective gaming. In Proceedings of the *9th Iberian Conference on Information Systems and Technologies* (pp. 1-6).

Noldus, L. P. J. J. (1991). The observer: A software system for collection and analysis of observational data. *Behavior Research Methods, Instruments, & Computers, 23*(3), 415-429.

Oracle. (n.d.). *Java Persistence API FAQ.* Retrieved from http://www.oracle.com/technetwork/java/javaee/persistence-jsp-136066.html

Ortiz de Guinea, A., & Webster, J. (2013). An investigation of information systems use patterns: Technological events as triggers, the effect of time, and consequences for performance. *MIS Quarterly, 37*(4), 1165-1188.

Ortiz de Guinea, A., Titah, R., & Léger, P.-M. (2013). Measure for measure: A two study multi-trait multi-method investigation of construct validity in IS research. *Computers in Human Behavior, 29*(3), 833-844.

Ortiz de Guinea, A., Titah, R., & Léger, P.-M. (2014). Explicit and implicit antecedents of users' behavioral beliefs in information systems: A neuropsychological investigation. *Journal of Management Information Systems, 30*(4), 179-210.

Ouwerkerk, M., Dandine, P., Bolio, D., Kocielnik, R., Mercurio, J., Huijgen, H., & Westerink, J. (2013). Wireless multi sensor bracelet with discreet feedback. In *Proceedings of the 4th Conference on Wireless Health* (pp. 6-14).

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45-77.

Peirce, J. W. (2007). PsychoPy: Psychophysics software in Python. *Journal of Neuroscience Methods, 162*(1-2), 8-13.

Pettit, J., Friedman, D., Kephart, C., & Oprea, R. (2014). Software for continuous game experiments. *Experimental Economics, 17*(4), 631-648.

Pfeiffer, J., Meißner, M., Brandstätter, E., Riedl, R., Decker, R., & Rothlauf, F. (2014a). On the influence of context-based complexity on information search patterns: An individual perspective. *Journal of Neuroscience, Psychology, and Economics, 7*(2), 103-124.

Pfeiffer, J., Meißner, M., Prosiegel, J., & Pfeiffer, T. (2014b). Classification of goal-directed search and exploratory search using mobile eye-tracking. In *Proceedings of the International Conference on Information Systems.*

Pfeiffer, J., Pfeiffer, T., & Meißner, M. (2015). Towards attentive in-store recommender systems. In D. Power & I. Lakshmi (Eds.), *Reshaping society through analytics, collaboration, and decision support* (pp. 161-173). Berlin: Springer.

Presentation. (2016). *Presentation: Precise, powerful stimulus delivery* (computer software). Berkeley, CA: Neurobehavioral Systems.

Randolph, A. B., & Jackson, M. M. (2010). Assessing fit of nontraditional assistive technologies. *ACM Transactions on Accessible Computing, 2*(4), 1-31.

Randolph, A. B., Borders, A. L., & Loe, T. W. (2013). Into the mind of the seller: Using neurophysiological tools to understand sales techniques. In *Proceedings of the 46th Hawaii International Conference on System Sciences* (pp. 425-429).

Riedl, R. (2013). On the biology of technostress: Literature review and research agenda. *DATA BASE for Advances in Information Systems, 44*(1), 18-55.

Riedl, R., & Léger, P.-M. (2016). *Fundamentals of NeuroIS: Information systems and the brain*. Berlin: Springer.

Riedl, R., Banker, R. D., Benbasat, I., Davis, F. D., Dennis, A. R., Dimoka, A., Gefen, D., Gupta, A., Ischebeck, A., Kenning, P., Müller-Putz, G., Pavlou, P. A., Straub, D. W., & vom Brocke, J. (2010a). On the foundations of NeuroIS: Reflections on the Gmunden Retreat. *Communications of the Association for Information Systems, 27*, 243-264.

Riedl, R., Davis, F. D. & Hevner, A. R. (2014a). Towards a NeuroIS research methodology: Intensifying the discussion on methods, tools, and measurement. *Journal of the Association for Information Systems, 15*(10), i-xxxv.

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 Quarterly, 34*(2), 397-428.

Riedl, R., Mohr, P. N. C., Kenning, P. H., Davis, F. D., & Heekeren, H. R. (2014b). Trusting humans and avatars: A brain imaging study based on evolution theory. *Journal of Management Information Systems, 30*(4), 83-114.

Schaaff, K., Degen, R., Adler, N., & Adam, M. T. P. (2012). Measuring affect using a standard mouse device. *Biomedical Engineering / Biomedizinische Technik, 57*(SI), 761-764.

Schneider, W., Eschman, A., & Zuccolotto, A. (2002). *E-prime reference guide.* Pittsburgh: Psychology Software Tools.

Seithe, M. (2012). *Introducing the Bonn Experiment System* (BoXS). Germany: University of Bonn. Retrieved from http://boxs.uni-bonn.de/boxs_seithe.pdf

Seuken, S., Parkes, D. C., Horvitz, E., Jain, K., Czerwinski, M., & Tan, D. (2012). Market user interface design. In *Proceedings of the 13th ACM Conference on Electronic Commerce* (pp. 898-915).

Shim, J., Varshney, U., & Dekleva, S. (2006). Wireless evolution 2006: Cellular TV, wearable computing, and RFID. *Communications of the Association for Information Systems, 18*, 497-518.

Tams, S., Hill, K., Ortiz de Guinea, A., Thatcher, J., & Grover, V. (2014). NeuroIS—alternative or complement to existing methods? Illustrating the holistic effects of neuroscience and self-reported data in the context of technostress research. *Journal of the Association for Information Systems, 15*(10), 723-753.

Teubner, T., Adam, M. T. P., & Riordan, R. (2015). The impact of computerized agents on immediate emotions, overall arousal and bidding behavior in electronic auctions. *Journal of the Association for Information Systems, 16*(10), 838-879.

Vance, A., Anderson, B. B., Kirwan, C. B., & Eargle, D. (2014). Using measures of risk perception to predict information security behavior: Insights from electroencephalography (EEG). *Journal of the Association for Information Systems, 15*(10), 679-722.

vom Brocke, J., & Liang, T. P. (2014). Guidelines for neuroscience studies in information systems research. *Journal of Management Information Systems, 30*(4), 211-234.

vom Brocke, J., Riedl, R. &. Léger, P.-M. (2013). Application strategies for neuroscience in information systems design science research. *Journal of Computer Information Systems, 53*(3), 1-13.

Warkentin, M., Walden, E. A., Johnston, A. C., & Straub, D. W. (2016). Neural correlates of protection motivation for secure IT behaviors: An fMRI exploration. *Journal of the Association for Information Systems, 17*(3), 194-215.

Appendix A: Architecture Details

We first present Brownie's architecture along the y-axis of Figure A1: its built-in tier and its customizable tier. The built-in tier is the platform's stable core. It is ready to use and requires no (or minimal) alterations while designing an experiment. The built-in tier comprises 1) the entity layer, 2) the data access object (DAO) layer and the components that define, 3i) the generic server, and 4) the generic client. The entity layer performs the role of object relation mapping to persist Plain Old Java Objects (POJO) in a database, and it is implemented using Java Persistence API (JPA). In addition to a relational specification that supports mapping Java objects to a database, JPA supports a rich query language that facilitates static and dynamic queries[13]. The corresponding project for the entity layer is the ExpJPA project (see Figure A2).

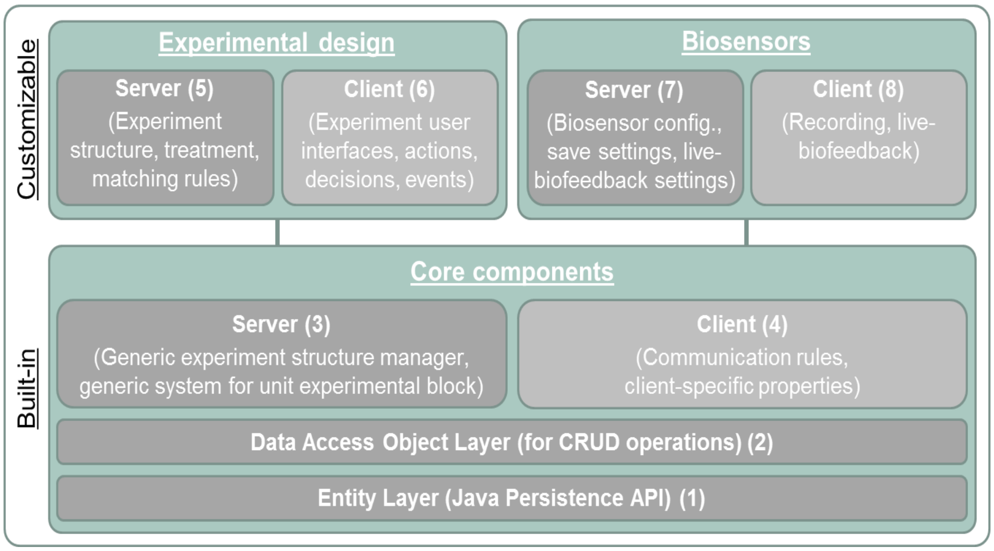


Figure A1. Brownie Architecture

The second layer of the built-in tier is the DAO layer, which serves as an abstraction to encapsulate all access to the data source. The DAO manages the connection with the data source to obtain and store data. As Figure A1 shows, through the DAO, one can access the entity layer from both the client- and server- sides (i.e., by all the components above it). The DAO layer applies the Data access object design pattern and is also located in the ExpJPA Project (see Figure A2).

The built-in tier finally contains two components to manage the structure of an experiment with the experiment structure manager and the experimental procedure on the server side (3) and the communication rules and client-specific properties on the client side (4). The experiment structure manager is invoked when the experimenter defines the flow of the experiment using Brownie's UI. Details about the treatments (interfaces associated with the treatment), sessions (session date, cohort size, membership and matching rules), experiment sequence information (number of periods, pauses, etc.) can be specified from the UI. We now explain each of these three aspects in detail.

The fundamental building blocks of a generic experimental design are implemented on the server-side middleware, namely: the institution and the environment. One can use the institution to define the experiment's behavior with respect to the start rules, the message rules, and end rules of the experiment. One can implement variations in the experiment's behavior across treatments in the same institution as well, and, hence, one can use a single class to define the flow across several treatments. Turning to the environment, it helps to define the primary experiment features and differences in experimental parameters across different treatments. For instance, for an experiment with two types of auctions and two levels of information granularity, there are four parameter combinations in a full-factorial design. One would then define the parameter sets for these four treatments in the environment class, whereas one

[13] As mentioned in the release notes of JPA, "[t]he Java Persistence API draws upon the best ideas from persistence technologies such as Hibernate, TopLink, and JDO. Customers now no longer face the choice between incompatible non-standard persistence models for object/relational mapping." (Oracle, n.d.).

would define the flow of the screens in in these four treatments in the institution class. An experiment always extends both classes, the institution and the environment, to define and implement specific scenarios.

Figure A2 depicts the underlying source code structure of Brownie. Four core component projects form the built-in tier: ExpServer, ExpClient, ExpCommon, and ExpJPA. Two projects form the customizable tier: Exp_Implementation for new experiments and ExpSensors project for existing and new biosensor configurations. These projects are available as a ready-to-use eclipse workspace along with definitions to the underlying dependencies of each project.

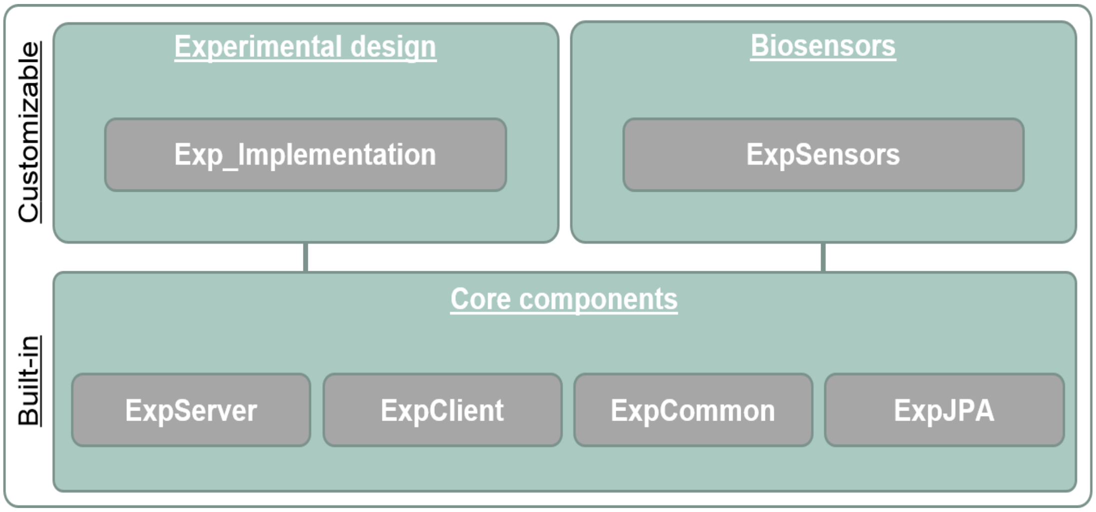


Figure A2. Source Code Project Structure in Brownie

Appendix A: Use Cases Implemented with Brownie

Table B1. Use Cases Implemented by Internal and External Researchers

Requirement: description	Expt. 1	Expt. 2	Expt. 3	Expt. 4	Expt. 5	Expt. 6
R1a: Individuals	x	x	x	x	x	x
R1b: Groups	x	o	x	x	x	x
R2a: Biosignals	x	x	x	o	o	x
R2b: Signal Quality Checks	o	o	x	o	o	x
R2c: Real-time signal processing	o	o	x	o	o	o
R3a: Support research on websites	o	o	o	o	o	o
R3b: Flexibility of data logging	x	x	x	x	x	x
R3c: Common programming language	x	x	x	x	x	x
R3d: Extensibility	o	x	x	o	x	x
R3e: Multimedia	o	x	x	o	o	x
R4a: Open source	o	o	x	x	x	x
R4b: Tutorials & support	o	o	x	x	x	x
R4c: Redistribution & replication	o	o	o	x	x	o
R4d: Questionnaires	x	x	x	x	x	x

Expt. 1, 2 and 6 were realized by members of the development team.
Expt. 3 and 5 were realized by other researchers in the same department of the development team.
Expt. 4 was realized in a different department of the development team's institute.
Expt.1 investigated the phenomenon of auction fever to study the influence of arousal.
Expt. 2 investigated the interplay of the constructs of cognitive workload and emotional arousal in auction bidding, using EEG, and ECG data.
Expt. 3 examined the influence of computer-agents, live bio-feedback and the IS constructs of current emotional state on individual human behavior in a financial trading context.
Expt. 4 focused on bilateral negotiations in a three-player setting concerning the allocation of durable bads (e.g., toxic or nuclear waste) (realized by Institute for Industrial Production, Karlsruhe Institute of Technology).
Expt. 5 was an oligopoly experiment that investigated decision making behavior in a simultaneous wholesale and retail competition context (realized at University of Passau and a field experiment conducted at Deutsche Telekom, Bonn)
Expt. 6 analyzed performance differences in serious games with regards to the IS constructs of arousal and workload.

Appendix C: Confirmatory Factor Analysis of IBM Usability Scale

We performed confirmatory factor analysis on a sample size of 57, including participants invited to both sessions. We performed exploratory factor analysis using maximum likelihood procedure on the data by testing the hypotheses that 1, 2, 3, and 4 factors are sufficient. The data loaded on three factors with a substantial increase in factor loadings: the confirmatory analysis yielded 117 degrees of model freedom with a chi-square of 144.39 (significant at the 0.1% level). The Tucker Lewis Index of factoring reliability was 0.953 and RMSEA index was 0.099. We examined correlation matrices to check that all factor loadings were greater than 0.7, and the item loadings were the same as IBM scale factors (system use, information quality, interface quality) except for one item. A comparison of different number (1-4) of factors (Table C1) shows that Tucker Lewis Index was greater than 1 for 4 factors, which denotes an over fitting of factors. Hence, for the usability study, we report results for three factors; the fourth factor represents the overall usability scale, which aggregates the above three factors.

Table C1. Confirmatory Factor Analysis for IBM Usability Scale

Model	Chi-square	df	BIC	Tucker Lewis Index	RMSEA
Single factor	326.53	152	-274.06	0.778	0.174
Two factor	240.95	134	-288.51	0.843	0.151
Three factor	144.39	117	-317.91	0.953	0.099
Four factor	95.64	101	-303.44	1.011	0.058

About the Authors

Anuja Hariharan is a Junior Consultant at Opitz Consulting gmBH. She received an undergraduate degree in Computer Science from the National University of Singapore, Singapore, and a PhD in Economics of Information Systems from the Karlsruhe Institute of Technology, Germany. Her research interests lie in the area of applying NeuroIS methods for studying the impact of emotions and cognitive workload in economic decision processes. Her work has been published in top international outlets such as *Electronic Markets, Economics Letters*, and *IEEE Transactions on Human-Machine Systems.*

Marc T. P. Adam is a Senior Lecturer in Computing and Information Technology at the University of Newcastle, Australia. In his research, he investigates the interplay of cognitive and affective processes of human users in human-computer interaction. He received an undergraduate degree in Computer Science from the University of Applied Sciences Würzburg, Germany, and a PhD in Economics of Information Systems from the Karlsruhe Institute of Technology, Germany. His research has been published in top international outlets such as *Business & Information Systems Engineering, Economics Letters, Electronic Markets, International Journal of Electronic Commerce, Journal of the Association for Information Systems, Journal of Interactive Marketing, Journal of Management Information Systems*, and*, Journal of Retailing.*

Verena Dorner is Assistant Professor at the Institute of Information Systems and Marketing at the Karlsruhe Institute of Technology and manager of the Karlsruhe Decision & Design Lab (KD2Lab). She received her PhD in Information Systems from the University of Passau. Her research on decision support, recommender systems and cognitive-affective aspects of decision making has been published in the *Journal of the Association for Information Systems, European Journal of Operational Research, Decision Support Systems, Economics Letters, Journal of Vocational Behavior*, and *Business & Information Systems Engineering.*

Ewa Lux is a PhD candidate at the Institute of Information Systems and Marketing, Karlsruhe Institute of Technology in Germany. She received a Bachelor's degree in Business Information Systems from the Baden-Wuerttemberg Cooperative State University in cooperation with the German Central Bank and a Master's degree in Information Engineering and Management from the Karlsruhe Institute of Technology. Her research interest centers on emotional processes in economic decision making in electronic markets. Her research has been published in top international outlets such as *International Journal of Electronic Commerce*, *Journal of the Association for Information Systems*, *Economics Letters*, and *Frontiers of Computer Science*.

Marius B. Müller is Executive Assistant at ITERGO. He received a Bachelor and a Master degree in Business Engineering and a PhD in Information Systems from Karlsruhe Institute of Technology, Germany. His research has been published in *Journal of Retailing* and *Proceedings of Hawaii International Conference on System Sciences*.

Jella Pfeiffer is Assistant Professor at the Institute of Information Systems and Marketing at the Karlsruhe Institute of Technology and manager of the Karlsruhe Decision & Design Lab (KD2Lab). She received her PhD in Information Systems and her Venia Legendi ("Habilitation") in Business Administration from the University of Mainz. She was visiting researcher at Harvard and the University of British Columbia as well as invited professor at the business school of the University of Lausanne (HEC). Her research interests include decision support systems in e-commerce and m-commerce, consumer decision making, Neuro Information Systems (in particular eye-tracking), human-computer-interaction and experimental research in the laboratory, the field and the virtual reality. Her work has been published, among others, in the *Journal of the Association for Information Systems*, the *European Journal of Operational Research*, and the *Journal of Behavioral Decision Making*.

Christof Weinhardt is director of the Institute for Information Systems and Marketing at the Karlsruhe Institute of Technology, Germany, director and co-founder of the Karlsruhe Service Research Institute and director of the Research Center of Information Technology. He received his Ph.D. in Economics from the University of Karlsruhe in 1989. Then, he focused his research on Information Systems at the Universities of Giessen and Bielefeld, Germany. With his academic background in industrial engineering, economics, and information systems, his research is on interdisciplinary topics in market engineering with applications in IT services, energy, finance, and telecommunications markets. He has published more than 150 articles and books and has received a number of awards for his research and teaching. His work has been published, among others, in *Energy Policy*, *European Journal of Operational Research*, *IEEE Transactions on Power Systems*, *International Journal of Electronic Commerce*, *Journal of Management Information Systems*, and *Transportation Science*.