Experiencing Software Landscapes using HCI in ExplorViz

Bachelor's Thesis

Matthias Möller

September 30, 2017

Kiel University
Department of Computer Science
Software Engineering Group

Advised by: Prof. Dr. Wilhelm Hasselbring
M.Sc. Christian Zirkelbach

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Kiel, 9. Dezember 2017

Abstract

The Brain-Computer-Interface (BCI) is an interface receiving the brain's signals and converting them into computer signals. Although the BCI's importance increased in the last years and most probably will increase further, most people have never heard of this device, because this device has yet not gained much place in our daily life routines, but this could change soon. At the moment portable, low-cost BCIs emerge from the market and open new opportunities in the context of interaction between human and machine, but these low-cost BCIs are mostly untested. Furthermore, there are barely approaches to implement the BCI as additional tool in an already existing software, yet.

Our working processes were inspired by gaining informations about the BCI as new possibility for task comprehensions. Therefore, we wanted to implement a BCI's functionality into the software tool *ExplorViz*. We worked with a low-cost BCI called *Emotiv Insight* and made in the beginning first experiences with the BCI as new interface. After we had designed our software, we implemented a plugin, which optionally extends *ExplorViz* by the possibility to use the BCI and, thereby, to use two different thoughts as additional navigation possibilities. In our implementation phase we received the chance of a collaboration with another bachelor thesis extending *ExplorViz* by a virtual reality (VR) concept. We combined our both implementations and evaluated our works in form of a study inspired in receiving informations about the user's acceptance. We divided the study in a VR and a BCI part.

In our BCI part we wanted to receive an answer to our research question *Will the user accept our implementation of the BCI in VR*. Therefore, we made three hypotheses. The first hypothesis regards the user's acceptance about the combination between VR and BCI as concept. The second dealt about the acceptance of a concept we implemented to give the user control over the BCI with help of the VR controllers and the third dealt about their feeling of being able to trigger at least one mental command to their satisfaction.

In the beginning of the study the participant received a directed training of approximating 72 second. Afterwards, we let the participants of our study perform 4 tasks by only using two different thoughts and the VR controllers. Finally, we pleaded the user to fill a check questionnaire designed in regard of our hypotheses. The answers of this survey proved our hypotheses and thereby our research question.

Contents

Chapter 1

Introduction

1.1 Motivation

The Brain-Computer-Interface (BCI) is a new device providing the opportunity to explore new paths of task comprehension in software engineering, but also shows new ways of interaction between human and computer. From the view of the Human Computer Interaction (HCI) we gained with the BCI a new interface connecting humans with computers. This technology is in an early phase of market integration and is not yet considerably implemented in our daily life routines. This early phase offers some interesting questions about the human's acceptance regarding this device and, thereby, the BCI's real additional benefits. While we found many approaches of integrating a BCI in very special fields like computer games or medical purposes, we have not found many integrations of BCI's as a simple additional tool in everyday software. Our goal is to describe an implementation of the low cost BCI *Emotiv Insight* into the software tool *ExplorViz* and, thereby, explore new ways of task comprehension.

Changing requirements, increasing workload, bad documentation or changed enterprise structures are natural side effects in long working processes. Accordingly, long-term developed software is influenced by these aspects too and has to evolve. As consequence these changes often result in a growing lack of code-understanding and bad documentation. [Zirkelbach 2017; Fittkau et al. 2015] *ExplorViz* analyzes software by using dynamic analysis methods. These methods enable software-engineers to analyze this long-term developed software and, thereby, they can find weaknesses and bottlenecks as long as the visualization is appropriate enough [Zirkelbach et al. 2015]. *ExplorViz* presents the analysis' result in form of a visualization. By interacting with the visualization the software-engineer can deepen his/her understanding of the software's structure and we want to know if this device can provide a new form of interaction.

1.2 Document's Structure

First, we describe in Chapter 2 the goals we try to achieve. In Chapter 3 we explain the foundations to understand our device and our further working steps. Afterwards, we discuss in Chapter 4 works related to our topic. Following, we explain our design in Chapter 5 and present its implementation in Chapter 6. Finally, we prove our created application in terms of usability in Chapter 7, draw conclusions from our work in Chapter 8 and close with Chapter 9 giving ideas of possible future work.

Chapter 2

Goals

We want to implement the BCI *Emotiv Insight* in *ExplorViz* as additional tool for task comprehension and finally test our implementation in terms of usability. Therefore we have following goals leading our working procedures.

2.1 G1: Identification of related Work and Technology

At first, we want to receive more knowledge about related work and technologies, which can inspire and alter the ways of our working process. On one hand we search for existing work we can use for our implementation, on the other hand we make assumptions on base of other works about our device's limitations.

2.2 G2: Design

We want to integrate a new device as tool into an already existing software. We have to decide, how we integrate our device and how it will influence our software and also for which tasks we want to use our device.

2.2.1 G2.1: Design of our Software

We want to add the BCI's functionality into *ExplorViz*. Consequently, we have to make decisions about the way, how we want to connect both with best possible performance and sustainability. At first we have to decide, where we want to implement our BCI in *ExplorViz*. Afterwards we have to decide, how our device and *ExplorViz* can communicate with each other. In addition we have to decide, how the user communicates with the device.

2.2.2 G2.2: Design of our Functions

We have to make general decision, how we want to use our device in the context of *ExplorViz*. Therefore, we have to regard the device's limitations and also find a meaningful function.

2.3 G3: Implementation

Regarding our design we implement the *Emotiv Insight* in *Explorviz*. We have to adapt aspects of our design, which are simply not realizable and find other opportunities.

2.4 G4: Evaluation

In the end we evaluate our software in consideration in the user's experience and it's ease. We are interested to see, if persons can control the BCI and how our implementation worked out. Furthermore, we want to receive the opinion of the user in terms of the device and our concept of using the device.

Chapter 3

Foundations and Technologies

3.1 The Brain

To understand our task's difficulty some information about the brain and its electrical nature are needed. The brain is the central part of our nervous system. It consists of more than 100 billion neurons, which interact through biochemical and electrical reactions [Burmaoglu 2002; Fichter 2002].

Every neuron has an amount of dendrites and one axon. The dendrites are small protuberances growing out of the cell core. Their functionality can be described as a receiver for electrical signals. The axon, a long extension of the neuron, holds the role of the sender. The axon of a neuron is connected with 10000 dendrites of others neurons. The connection between axon and dendrite is the synapse. Synapses are filled with neurotransmitters.[Kaku 2014; Fichter 2002]. We can shortly describe the communication between axon and dendrites as followed: By the appearance of an action potential with enough strength a polarization occurs in the axon. This polarization generates an electrical charge. The electrical charge is transmitted with help of chemical processes in the synapse to the dendrites. In turn the dendrites transmit the signal to the own cell core [Fichter 2002].

3.2 EEG Signal

The physics equipped us with methods to measure brain activity. These methods enables us to find more insight into the brain and are the foundation of the device we use. As mentioned before, the brain works with electrical and chemical processes. For about a century we have known about the electrical nature of the human nervous system, what inspired us to measure brain activity in form of electrical charge.

The Electroencephalography (EEG) is based on the works of Hans Berger out of the year 1924 and represents the scalp's surface potential. We record the EEG by affixing an array of electrodes to the head and measuring the voltage between pairs of electrodes in its frequency and amplitude [Ochoa 2002]. To analyze EEG signals is a science in itself and concludes many process like filtering, amplifying and visualizing the received data, but of course we find a deeper meaning in EEG.

Table 3.1. The exact categorization of wave pattern is still very vague, but they clearly represent the mental state of a person[Ochoa 2002].

Wave Group	Frequency	Amplitude
Alpha	8-13 Hz	8-13 μV
Beta	13-30 Hz	5-30 μV
Delta	0.5-4 Hz	variable
Theta	4-7 Hz	>20 μV

3.2.1 EEG Waves

We gain much knowledge about a person emitting the EEG Signal by observing its frequency and its amplitude. We can categorize these EEG signals by their amplitude and frequency in so called wave groups and draw thereby conclusions about the mental state of its owner. There are at least four wave patterns of interest, although the amount of interesting wave groups is not yet completely settled [Ochoa 2002; Vallabhaneni et al. 2005].

Alpha: The alpha wave is the most prominent wave. This wave pattern indicates an relaxing mental state, although the subject is not tired or asleep. Of course the pattern is not yet completely unscrambled, but at least we are assure, that they are produced by the subject's inattention or even some kind of mindless state [Graimann et al. 2010; Ochoa 2002].

Beta: A beta wave goes along with active thought process, visual fixing and concentrating. In intense mental activity the beta waves can reach near 50 hertz [Ochoa 2002].

Delta: These waves indicate deep sleep. In waking state these waves indicate physical brain damage. It is easy to confuse jaw- or neck-muscles with Delta waves because of the low frequency [Ochoa 2002; Shaker 2006].

Theta: They are normally measured in stress situations including frustration and disappointment, but they are also associated with meditation and creative inspiration [Ochoa 2002; Shaker 2006]. It's more likely, that they occur during childhood [Shaker 2006].

3.2.2 Sampling

There are two prominent ways to sample EEG signals. An affixing of an array of electrodes on the head is called noninvasive, whereas the affixing into the gray matter of the brain is called invasive. The invasive sampling decreases risks and are easier to accomplish, while the invasive method is more accurate [Vallabhaneni et al. 2005].

EEG signals represent the electrical potential in the cortex, but the brain works with only 20 Watt and thereby the signals are very weak [Kaku 2014]. Consequently, we have to filter and process signals, but also an accurate sampling is essential. The electrodes have to be able to receive the signals most clearly. Therefore, we often use in combination of noninvasive sampling some kind of gel for better conductibility. Additionally, signal noises can occur and thereby we have to filter these signals.

More electrodes and consequently more samples indicate more accuracy. More samples make it easier to differ between different thoughts, because we can observe more brain areas and thereby can better discriminate signals. Standard medical EEG devices generally have 19 electrodes or even more, while most of the high-resolution EEG systems consist out of 64 or 128 electrical sensors. In spite of the importance of the amount of electrodes, its arrangement seems to have big influence on the results, too [Babiloni et al. 2004; Lang 2012].

3.3 Brain-Computer-Interface

The Brain-Computer-Interface (BCI) is an interface between brain and computer. Although the BCI is a tool like the mouse or the keyboard, it is much more difficult to handle [Vallabhaneni et al. 2005]. The BCI works with the user's brain activity often measured by EEG signals. To use the BCI properly the user has to be trained in controlling his own mind and in most cases the program has to be adapted onto the user [Burmaoglu 2002; Graimann et al. 2010; Renard et al. 2010].

At first a BCI needs training. In this training the BCI collects and expands a training set by sampling measurements of brain activity. By using the training set, the BCI can compare new signals and is afterwards able to recognize already learned thoughts again. Hence, a user can train thoughts as mental commands, which trigger the device and to get the device to do further actions.

3.4 The BCI Emotiv Insight

We work with a BCI called *Emotiv Insight*[1]. *Emotiv Insight* works with 5 EEG sensors and 2 reference sensors. It is produced and designed by *Emotiv Systems*. *Emotiv Systems* is specialized in making low cost BCIs for gaming and applications. Additionally our device belongs to the group of low cost, portable BCIs [Powell et al. 2013].

3.4.1 Community SDK

We work with the *Community SDK* of *Emotiv Systems* providing users to write own applications without needing much knowledge about the EEG signals. Originally, the *Community SDK* was written in C, but by including some libraries, which can be downloaded from a GitHub repository[2], we can also work with Java. These libraries work with Java 1.6, because much functionality is based on a pointer structure, which is not longer supported by Java

[1] `https://emotiv.com/insight.php`, as of May 2017
[2] `https://github.com/Emotiv/community-sdk`, as may 2017

1.7 or higher. We want to give some needed information being essential to understand the device. The *Community SDK* offers five different libraries, but only three are essential for a proper usage:

EdkErrorCode.java: Many functions return a so called *EdkErrorCode*. EdkErrorCode.java helps to understand these Codes and thereby informs the user about a success or a failure.

Edk.java: Almost every function accessing the device is located in our library Edk.java. We use this device to access control over its working flow and hereby we can start training sessions for example. Furthermore, we use events called *EmoEngineEvents* to acquire information about our device.

EmoState.java: This library supplies us with functions filtering wanted data out a so called *EmoState*. While we used our Edk.java library to achieve *EmoEngineEvents*, we use the EmoState.java library to interpret the *EmoState* and filter for example information about the facial Expression. We do not see a completely understandable structure in the usage of *EmoState* and *EmoEngineEvent*, but we see the *EmoEngineEvent* as some kind of upper level event defining our *EmoState*.

3.4.2 Mental Commands

The *Emotiv Insight* is able to provide information about a person's mental state as well as the facial expression. Because we use the *Community SDK*, we can not access pure EEG signals, and are limited to some functions. The user is able to train some thoughts as mental commands, which will be analyzed and processed by the BCI. The training session is shown in Figure 3.1. By telling the device, which mental command we want to train and indicating the training start, the device starts recording EEG signals. Afterwards the device gives the user the opportunity to accept the training or to reject the collected training data.

If the training was successful and accepted by the user, the BCI will be more or less successful in recognizing this thought as the mental command and differentiating this thought from other learned mental commands. If a thought is recognized, the machine is able to inform the user by passing a mental command in form of a number.

The device's way of decoding the mental commands needs some explanation. The first mental command will be decoded with $2 = 2^1$, the second with $4 = 2^2$ and the third with $8 = 2^3$. The device simply uses the binary number system for decoding the mental commands. Additionally, *Community SDK* has its own notation of the mental commands. The first Command is called MC_PUSH and the second is called MC_PULL. These are just names, which are written down in the library EmotState.java.

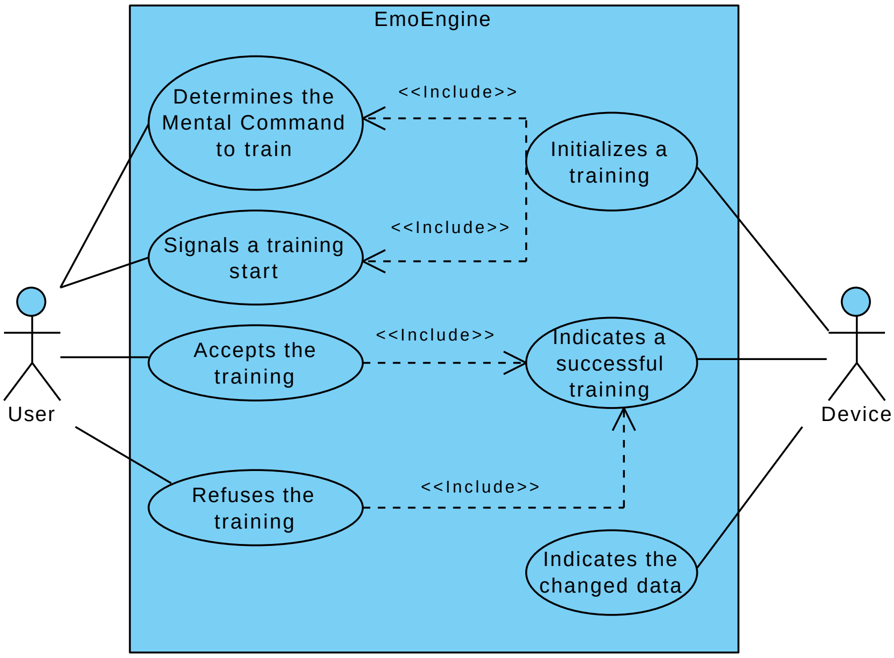

Figure 3.1. Use case diagram for training sessions

3.4.3 The Communication with the Device

The device's nature allows us no insight into its internal processing and gives us only the opportunity to delegate tasks. Furthermore, the machine only accepts one request per time, else it crashes.

The device writes into pointers requested information. Afterwards the user can process and interpret these informations. We can say, the device uses events to show the availability of interactions. For example Figure 3.2 shows that by achieving the next *EmoEngineEventType* from the *EmoEngineEvent* you can get insight into the device's working state. When the device ended a training, it answers with a *MentalCommandEventType*. When the device has recognized a thought as mental command, you are able to ask for a *EmoState* and extract information about the mental command by processing the *EmoState*.

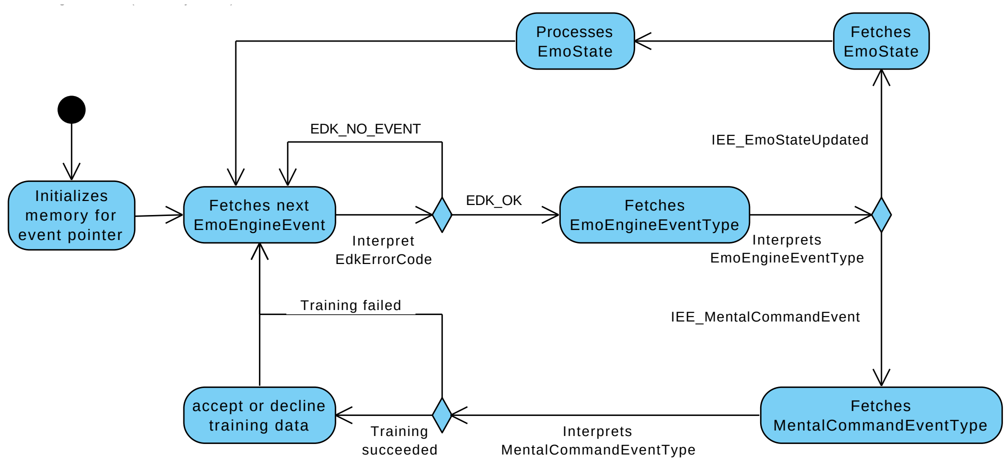

Figure 3.2. Activity diagram showing one way to work with events

The same is shown in Listing 3.1 as code example. In line 7 we ask for an update of our *eEvent* and fetch simultaneously the *edkErrorCode*. If the device offers new information, we can fetch the *EmoEngineEventType* demonstrated in line 8. On basis of the *EmoEngineEvent-Type* we can determine further working processes. If the *EmoState* provides new information (line 10), we gain knowledge about one's mental state or facial expression. Furthermore the device use the *EmoEngineEventType* to demand actions from the user. If we receive a *MentalCommandType*, the device implies a training start or end of a Mental Command.

Finally, we want to accent that the event handling of our device can become much more complex. Listing 3.1 demonstrates, how big a code, which handles events appropriately, can become. In our code we have have no functionality and neglect aspects like a permanently updating or reactions to our user. A main point in our work will be the task to combine permanent updates, event handling and GUI receiving requests from the user.

3.5 WebSocket Protocol

The WebSocket protocol enables a connection between a client to a remote host and is based on TCP. This connection is special in the fact that it's unidirectional and runs not trusted code. The protocol consists out of an opening handshake and is followed by basic frame messages. Generally, this protocol aims for a communication between a browser and a server. This enables the constant update of browser-based applications, which need data from a server [Fette 2011]. More information can be received on the website of the Internet

Listing 3.1. code-example to manage device

```
//connects the device
Edk.INSTANCE.IEE_EngineConnect("Emotiv␣Systems-5");
//allocates memory for writing
Pointer eEvent = Edk.INSTANCE.IEE_EmoEngineEventCreate();
Pointer eState = Edk.INSTANCE.IEE_EmoStateCreate();
int edkErrorCode = Edk.INSTANCE.IEE_EngineGetNextEvent(eEvent); //actualizes the
    EmotivEvent
if(edkErrorCode == EdkErrorCode.EDK_OK.ToInt()){ //new data
  int emotivEventType = Edk.INSTANCE.IEE_EmoEngineEventGetType(eEvent); //gets the
       EmotivEventType
  //new data recorded by device
  if(emotivEventType == Edk.IEE_Event_t.IEE_EmoStateUpdated.ToInt()){
    Edk.INSTANCE.IEE_EmoEngineEventGetEmoState(eEvent,eState); //altualizes
        EmotivState
    //receiving the needed information, if blinking
    int isBlink = EmoState.INSTANCE.IS_FacialExpressionIsBlink(eState);
    //receives information about current Mental Command
    int mentalCommandType = Edk.INSTANCE.IEE_MentalCommandEventGetType(emotivEvent
        );
    int currentMentalAction = EmoState.INSTANCE.IS_MentalCommandGetCurrentAction(
        eState);
    //more code

  }else if(emotivEventType == Edk.IEE_Event_t.IEE_MentalCommandEvent.ToInt()){
    //a training started, were completed or else
    //more code
  }else if(emotivEventType == Edk.IEE_Event_t.IEE_FacialExpressionEvent.ToInt()){
    //more code
  }
}
//Disconnects the device and free the memory
Edk.INSTANCE.IEE_EmoEngineEventFree(eEvent);
Edk.INSTANCE.IEE_EmoStateFree(eState);
Edk.INSTANCE.IEE_EngineDisconnect();
```

Engineering Task Force (IETF)[3].

3.6 ExplorViz

ExplorViz is a web-based, live trace visualization tool concluding backend and frontend and aims for solving system comprehension tasks fast and accurately for large software landscapes by using hierarchical abstractions [Fittkau et al. 2017]. Especially, enterprise information systems belongs to *ExplorViz* target group.

ExplorViz uses dynamic analysis methods working on system executions for analyzing software. Thereby, we can analyze legacy systems and reverse engineer their software structures. By using dynamic analysis methods *ExplorViz'* working process can be divided in two consecutive phases. At first the application has to monitored and afterwards it has to apply analysis and visualization on the monitored data [Zirkelbach et al. 2015]. The monitoring provides *ExplorViz* with up-to-date informations being consistent with the actual applications and information systems [Fittkau et al. 2015; 2013]. *ExplorViz* visualizes the analyzed data in two different abstract forms.

The figure shown in Figure 3.3 is a 2D model and abstracts the relation between systems, applications and communications in form of a landscape. A system consists out of nodegroups, which then again are defined by nodes. *ExplorViz* uses for the landscape's design a flow-based layout, which means that the communications will also flow from the left source to the right target [Fittkau et al. 2015].

An application shown in Figure 3.4 is characterized by its components and has its own visualization in form of a 3D model. While our 2D landscape is designed with a flow-based layout our application is visualized with the help of a 3D city-layout. The object's height can be referenced with the actual number of instances Fittkau et al. 2015.

3.7 Ember.js

Ember.js[4] is a JavaScript framework for building web applications designed for the web-browser as platform. Consequently, *Ember.js* is used for frontend-design and equips developer with many useful features, like a well understood object-oriented pattern enabling users to extend own classes, the writing of own model classes and complex router configurations [Cravens and Brady 2014]. *Ember.js* is highly customizable and the integrated plugin-structure allows an easy inserting of additional functionalities.

[3] `https://tools.ietf.org/html/rfc6455`, as may 2017
[4] `https://www.emberjs.com/`

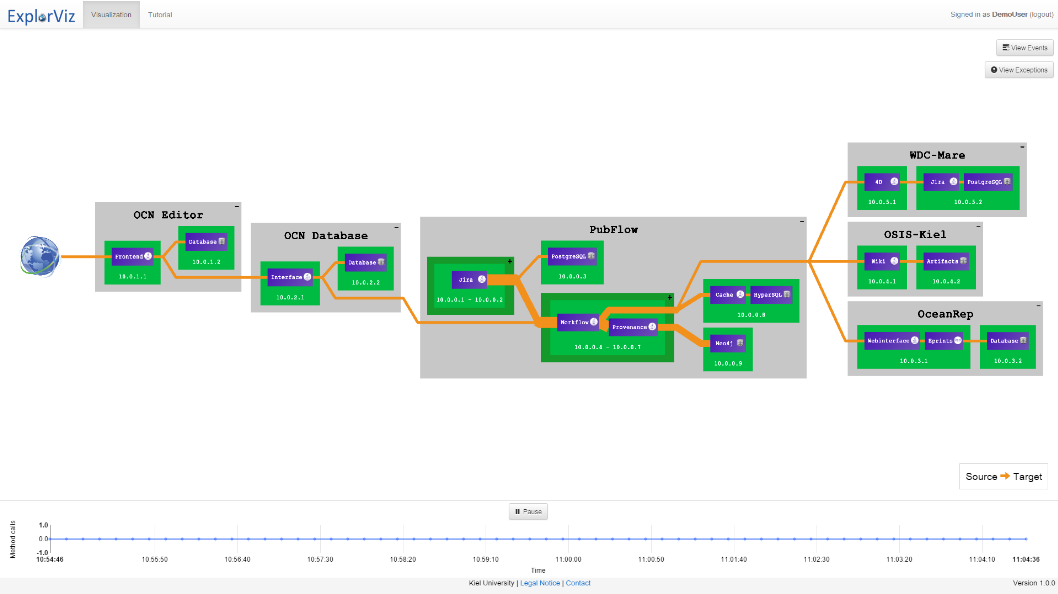

Figure 3.3. An visualization of a landscape

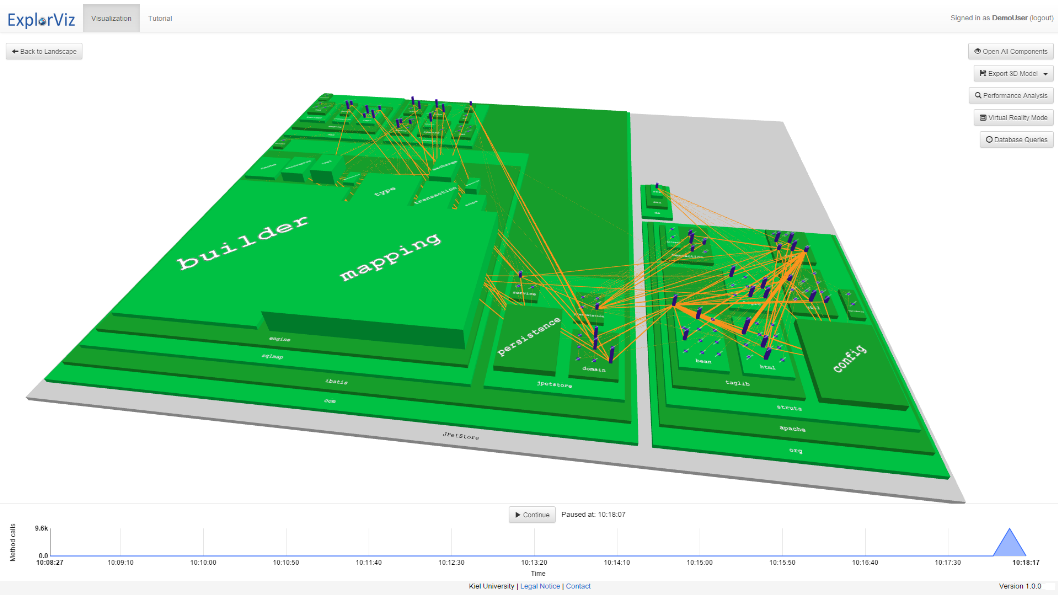

Figure 3.4. An visualization of an application

Chapter 4

Related Works

We try to integrate the low cost BCI *Emotiv Insight* into *ExplorViz* with usage of the *Community SDK* of *Emotiv Systems*. Unfortunately, we did not find many works being directly related to our goal. Hence, we will discuss works, which are distinctly related, and we try to abstract our achieved knowledge on useful aspects.

Taylor and Schmidt made an empirical evaluation of the *Emotiv Epoch* for the detection of mental actions [Taylor and Schmidt 2012]. In their work the device was able to detect a triggered mental command with an accuracy of about 69,7 % and the BCI was able to differ two mental commands with an accuracy of 87,5 %. Furthermore, they observed an increasing of the systems sensitivity by 35% from the first to the third training. The *Emotiv Epoch*[1] is another BCI from the company *Emotiv Systems*. While our device only has 5 EEG sensors and two reference sensors, the *Emotiv Epoc* has 14 EEG sensors and 2 reference sensors. Consequently, we can not imply the *Emotiv Insight's* accuracy on base of the *Emotiv Epoc's*, but at least we see a chance to receive an appropriate result in differing two different Mental Commands. Furthermore, we want to regard their observations about the increasing of the device's accuracy after three training sessions.

Levicán et al. worked with the *Emotiv Insight* in context of the creation of a music application.[Levicán et al. no date] They mentioned, that they used a SDK from *Emotiv Systems*, but they did not mentioned which SDK exactly. We would deduce, that they do not use the *Community SDK*, because they were able to access EEG-data. Independently, they expressed their suspicion the device would lack in reliability and robustness. Indeed, this would be an important factor, which could danger our project.

Powel et al. made a study evaluating low cost BCIs, which are emerging from the market [Powell et al. 2013]. While we find many studies about the BCIs used in science, we do not find many informations dealing about low cost, portable devices. Powel et al. worked with the *Emotiv Insight* and they directly worked with the EEG-signals. They preprocessed their signals to remove noises and artifacts. Thereafter, they disjointed their signals and applied three different feature extraction methods. Finally, they used a support vector machine as classifier and used a error-correcting output code as multiclass model. Depending on the feature extracting method, they were able to achieve a recognition accuracy with about 95% [Powell et al. 2013]. We work with the *Community SDK* and, thereby, have no access to EEG signals, so we are not able to influence the ways our signals are being processed, but at

[1] https://www.emotiv.com/product/emotiv-epoc-14-channel-mobile-eeg/

least we see the chance that *Emotiv Insight* show an adequate accuracy depending on the machine learning algorithm our *Community SDk* uses.

By these three studies we draw the conclusion that our device could probably be able to discriminate thoughts adequately, but we shall better expect a low accuracy. We could not find a work similar to our task and, thereby, an own implementation is unavoidable.

Chapter 5

Design

We want to implement our BCI's functionality into *ExplorViz*. Therefore, we have to make general decisions regarding the way, how we implement our software and how we implement our BCI's functionality. We want to discuss general design decision consequently, which represent the guidelines of our integration.

5.1 Design of our Software

At first, we have to decide, where we want to integrate our software. *ExplorViz* consists out of back- and frontend. While our backend analyses our software and builds all components defining our software landscape, our frontend has the task of layouting and visualizing the landscape. We decide to integrate our BCI by using our frontend, because we see no advantage to let our backend process data, which shall only influence our visualization in the frontend. Our frontend's platform is the browser and, consequently, we see the possibility to integrate our software into the browser as some kind of browser-internal application or we outsource some of the working processes in form of a browser-external software. These four aspects are fundamental to understand our decisions regarding this matter:

1. the device only accepts one request per time, else it crashes simply.
2. the device constantly adds its measurements in some kind of query. Afterwards we can access these informations.
3. we can request data, but we always receive oldest data in the query.
4. software integrated in the browser is limited by the browser's limitations.

Considering these aspects our design suggests a browser-external software processing most of the data and communicating with the browser-internal part via WebSocket connection and with our device via bluetooth shown in Figure 5.1.

Our BCI independently and constantly adds more and more data into the query and, consequently, the amount of data in the query becomes bigger and bigger. One the other hand, on our demand the device does not provide us with the most actual data set in the query but the oldest. Hence, we have to process our data in the same speed, else we fall back

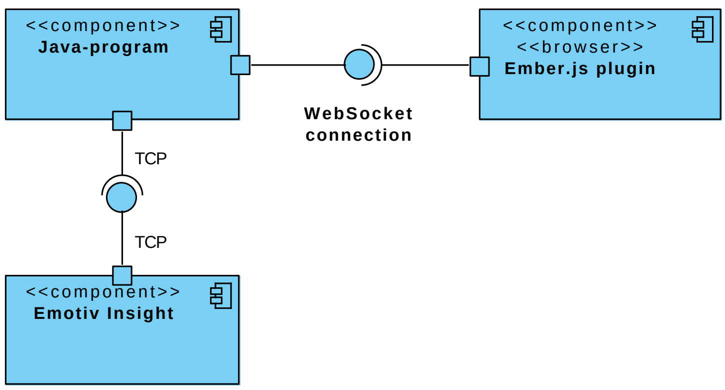

Figure 5.1. Component diagram describing our envisioned software approach

and, thereby, we loose the context to the real-time data. Considering this racing situation between our application and the device, we see the performance as the most crucial part to ensure the currency of our data. Furthermore the browser is limited in its access to computer-based resources and is thereby not as efficient as browser-external applications. A browser-internal application would probably not have the performance to process the device's data fast enough. Additionally, web applications do not support multi-threading or multi-processing naturally. As consequence we probably would not achieve our aimed thread-handling-management and would limit the user in his/her usability. By using a browser-external Software for processing data we force two phases on the user:

1. The user has to train his/her wished mental commands. Therefore he/she is forced to use the browser-external application.

2. The user has trained his/her mental commands and he/she wishes to us these in the application's browser-internal parts. Therefore he/she does not need the GUI of our browser-external application.

These two phases offer the opportunity to minimize the browser-external application's work based on the actual phase and therewith we can reduce the possibility to loose the context to the real-time data.

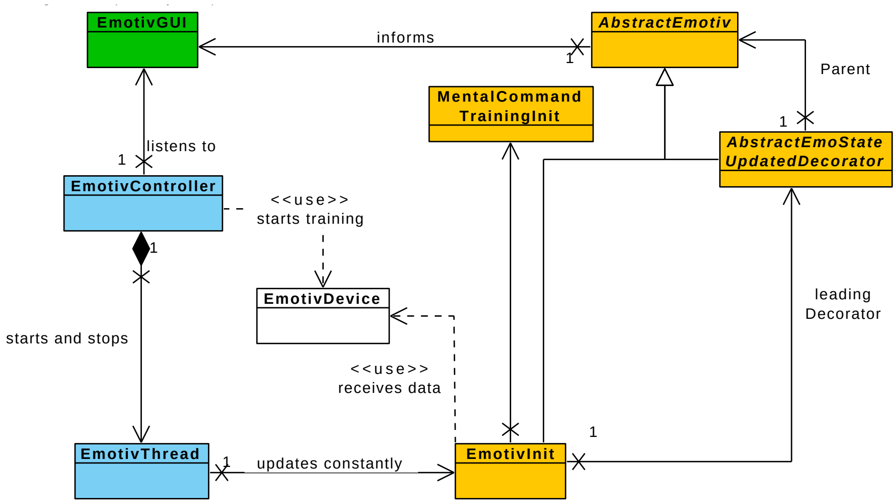

Figure 5.2. Class diagram showing our approach for our browser-external Java-code

5.1.1 Design of our Browser-External Software

Figure 5.2 shows our design of the browser-external software. We use an adapted version of the model-view-controller (MVC) software architecture pattern. The MVC pattern is described by the usage of a controller reacting to the users actions, a model representing the needed informations our software demands and a view visualizing the model's data.

Controller

Our controller is shown in blue. We divide our controller into the *EmotivController* and the *EmotivThread*. The *EmotivController* receives requests from the *EmotivGUI* and is able to start and stop the *EmotivThread*. While the *EmotivController* processes the user's requests, the *EmotivThread* shall handle the device's newly processed data by actualizing the model constantly.

Model

AbstractEmotiv, *EmotivInit*, *MentalCommandEventTypeInit* and the *AbstractEmoStateUpdat-eDecorator* represent our model shown in orange. This architecture is inspired by the decorator pattern and event-handling-management necessary to operate the device shown in Figure 3.2. The *EmotivInit* delegates tasks either to the *MentalCommandEventTypeInit* in case of a finished training or to one of the *AbstractEmoStateUpdateDecorator* in case of new processed data. It is possible to connect more *AbstractEmoStateUpdateDecorators* and thereby to change the program flow. We could also delete decorator, which are not needed, and reduce consequently the tasks our application has to make. In this context, we hope to increase the flexibility of our program flow, too. A programmer should be able easily to write new instances of *AbstractEmoStateUpdateDecorators* and to exchange parts.

In our design we plan to write a *WebSocketDecorator*, which processes the mental commands detected by the device. The *WebSocketDecorator* sends the mental command to our browser-internal part with help of the WebSocket protocol.

View

The GUI shall provide the user with every necessary information needed to work with the device. We do not want to give a user more information than he/she needs. Furthermore the user shall be able to influence our controller by interacting with the GUI. In the case of a started training session the *EmotivController* has to stop the *EmotivThread*. Afterwards we can initialize training sessions without crashing the device.

5.1.2 Design of the Browser-Internal Software

We want to include our BCI's functionality into the *ExplorViz* frontend. *ExplorViz* is based on *Ember.js* and provides us with many features. Therefore, we work with *Ember.js'* plugin structure. Accordingly, our BCI plugin will not be implemented as standalone application but needs to be implemented into the *ExplorViz* main application. We see more comfort to maintain our plugin and *ExplorViz* severally.

Additionally, we have to inform our user about the connection to the browser-external part of our software and provide a way to establish the WebSocket connection in case of a missing connection. We plan to use a button for both purposes. While clicking the button shall activate the connection, the button's color shall represent the connection's state. A green color shall symbolize an active and a gray button an inactive state of our connection.

Further, we see a problem in influencing the software's visualization with our plugin. *Ember.js* has a strict hierarchy regarding the communication of its elements and it is not desired to let the elements communicate freely. By our requirements, our browser-internal part shall be a plugin and is thereby limited in accessing the main application. We have to find some way to let the newly added components of our plugin communicate with our already written main application. We see this task as the most essential and challenging task in our implementation of the browser-internal part. Our first approach is the usage of a service. A service is an *Ember.js* class, which is a singleton and which is accessible from all objects in the *Ember.js* application. We could use this service as some kind of communication service.

Finally, we want to discuss our approach of the BCI-mode. We work with a device, which accuracy is mostly unclear and untested, but we already expressed our doubts in Chapter 4. A low accuracy would lead in wrong detections of mental commands and thereby could result in wrongly triggered functions. It is for this reason that we want to provide the user with the possibility to decide the time frame, when the BCI's detections have an actual effect on the landscape. With the key sequence ctrl+b the user is able to activate the BCI-mode and thereby to allow the device's influences. A further ctrl+b would lead to a deactivation of our BCI-mode. We choose ctrl+b as key sequence, because we wanted to ensure that the user does not activate the BCI-mode unintended and, furthermore, we did not find any information that it could be a short cut used in browser. Additionally, we think of controlling the BCI-mode as a good method for mentally preparing oneself. A user can prepare her-/himself in evoking his/her mental command. We think this is a good chance to reduce the emergence of unintended commands.

5.2 Design of our Mental Commands

For the purpose of our evaluation we want to find two thoughts triggering our BCI. These mental commands shall affect our application and shall be triggered as easy as possible. To maximize the devices accuracy in recognizing a mental action, we choose our two mental actions in consideration of the EEG wave characteristics explained in Section 3.2.1. Table 3.1 shows that beta waves corresponds with 13-30 Hz frequency and a 5-30 μV amplitude, while alpha waves are in the area between 8-13 Hz frequency and an 8-13 μV amplitude. We chose these waves to make the discrimination between the two mental commands as easy as possible. Rowland et al. made an evaluation in the year 1985 that indicates that alpha waves corresponds with mental demand and beta waves corresponds with emotional and cognitive processes [Rowland et al. 1985]. Consequently, alpha waves can be produced by closing the eyes and relaxing, whereas beta waves can be produced by mental work. Other EEG waves are not practicable to use. It would be difficult to imitate delta waves or theta waves, because they indicate deep sleep or mental stress. We see following difficulties in choosing a relaxing state as one of our mental commands:

1. the relaxing state will be difficult to be evoked
2. we will not find some useful function for a relaxing mental state in our application.

Although we see these dangers clearly, we hope to achieve an appropriate differentiation and thereby a better usability regarding the device's possibly bad accuracy.

Furthermore we have to make some decisions regarding our mental command's functionality in *ExplorViz*. Our choices of useful functionalities are limited in the context of a 2D visualization and so we decided to use the zoom for our evaluation's purposes. The zoom out is our choice for relaxation and the zoom in for concentration.

Chapter 6

Implementation

We want to discuss our software's implementation, but first we have to explain an aspects, which will influence our further work from now on.

6.1 VR collaboration

In our implementation phase, we received the chance for a collaboration with another bachelor thesis, which tries to implement a three-dimensional representation of a software landscape in *ExplorViz* in the context of virtual reality (VR). The other bachelor thesis also implements their software as *Ember.js* plugin. From now on we will refer this *Ember.js* plugin as *VR-plugin* The VR group uses VR glasses to simulate a virtual room containing an *ExplorViz* software landscape in form of a 3D visualization shown in Figure 6.1. We also see in this picture the two VR controller, which are used in the VR-mode.

We see many benefits in the integration of a BCI in a three-dimensional room. For

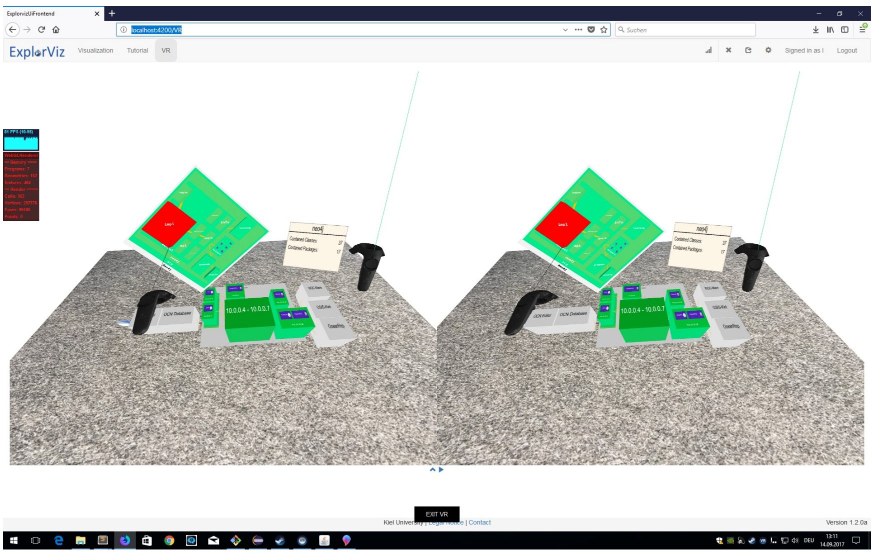

Figure 6.1. Visualization of a software landscape in VR

example we hope to increase the ease to evoke a mental command by concentrating on a 3D object. Furthermore, we see also possibilities to use the VR controller to strengthen one imagination.

6.2 Implementation of our Software

We implemented our browser-external part as Java 1.6 standalone application and our browser-internal part as *Ember.js* plugin.

6.2.1 Browser-External Software

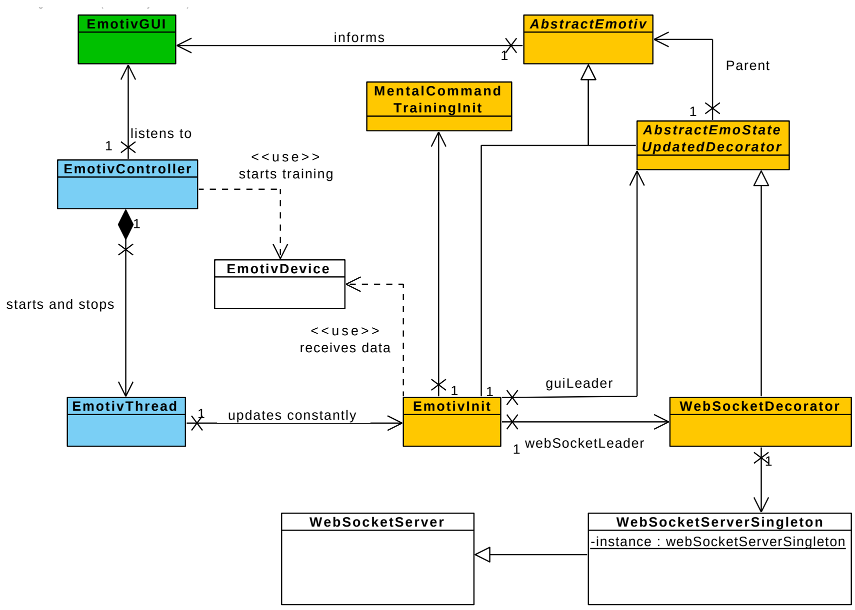

Figure 6.2. Class diagram showing our implementation based on the MVC pattern.

We implemented our MVC pattern by using *Emotiv System's Community SDK* and Java 1.6. Our model on base of the decorator pattern has proven itself as a comfortable way to expand or decrease our application's functionality. Our implementation is shown in Figure 6.2.

Controller

Our controller is shown in blue color in Figure 6.2. The *EmotivController* gives users the needed control over the device's working flow by commanding our *EmotivThread*. In Listing 6.1 we see, how our controller induces the start of a training session. At first our controller stops our permanently updating *EmotivThread*, because our device can not handle more than one request per time. Afterwards we give information about the mental command we want to train. We use our Edk.java library to access the device and also to receive information about the mental command we want to train. In this case we want to train our neutral mental command, which is used as some kind of reference to differ all other mental commands. Thereafter, we start our training in the same manner. 4096 is the default user-ID our device uses, if we are not using a specific profile. Finally we start our *EmotivThread* again.

Listing 6.1. Example code for starting a training in *EmotivController*

```
this.emotivThread.stopWorking();
Edk.INSTANCE.IEE_MentalCommandSetTrainingAction(4096,
  EmoState.IEE_MentalCommandAction_t.MC_NEUTRAL.ToInt());
Edk.INSTANCE.IEE_MentalCommandSetTrainingControl(4096,
  Edk.IEE_MentalCommandTrainingControl_t.MC_START.getType());
this.emotivThread.continueWorking();
```

Model

Our model is shown in Figure 6.2 in orange color and contains our basic ideas of the design with minor adjustments. Our *EmotivInit* has now two different leading decorator objects. The one leader is responsible to actualize our GUI, while the other is communicating with our browse-external part over the WebSocket connection. These two decorator classes are a result from our two different phases we have explained further in Section 5.1. While the user trains his/her mental commands, our *EmotivInit* delegate the update-tasks to its *guiLeader*. The *webSocketLeader* is only used for processing and sending mental commands. *WebSocketServerSingleton* is characterized by its singleton nature and extends the *WebSocketServer* class. *WebSocketServer* is provided by the library *Java-WebSocket*[1] and provides the whole logic we need for our WebSocket connection compatibe with Java 1.6. Furthermore our *WebSocketDecorator* makes use of this singleton to send processed data to our browser-internal application, if certain criteria are met. Our *WebSocketDecorator* only sends, if he detects a mental command twice in a row with a certain trigger strength. We added these criteria, because we observed in ourself that we were able to trigger one mental command with much less effort than the other. This had a frustrating effect on us. The user can decide the needed trigger strength in the GUI.

[1] https://github.com/TooTallNate/Java-WebSocket

View

Our model is shown in Figure 6.2 in green color. The *EmotivGUI* provides users an opportunity to either train mental commands or to activate the sending on the WebSocket connection.

In our GUI the user can train two mental commands and a neutral mental state. The neural mental state gives our device some kind of reference for differing mental commands. Although *Emotiv Insight* uses MC_PULL and MC_PUSH as appellation for the two first mental commands, these names only have a symbolic nature and the user can train every thought he/she wants. If we start a training, the GUI will hint the actual training start. Afterwards the user gets a chance to reject or accept the training.

6.2.2 Browser-Internal Software

Our browser-internal software was realized as *Ember.js* plugin. The user can install the plugin in the *ExplorViz* main application and thereby add the BCI's functionality. Our browser-external application uses an additional plugin called *ember-websockets*[2]. *Ember-websockets* allows us to connect, to listen and react to messages sent on a WebSocket connection.

The Interplay between Service and Button

In Listing 6.2 we can see parts of the service we use to handle the WebSocket connection.

Our variable *websockets* in line two refers to the service provided by the plugin *ember-websockets*. We only use *websockets* to start listening on a socket, if possible. In line 9 our service creates a reference to listen on socket 1313. Afterwards we add listeners on this reference, which are triggered, when certain events occur. While *openMessageHandler* and *myCloseHandler* change *status*, when a socket connection is opened or closed, our *myMessageHandler* handles arriving messages and is responsible to trigger functionalities.

The user is able to open the WebSocket connection by clicking on a button being located in the right upper corner of the website. Our button is an *Ember.js* component and we can see code parts in Listing 6.3. Line two shows the injection of our service described in Listing 6.2 and afterwards we can use *emotivWebsocket* to access its functionalities.

Our *classNameBindings* is an *Ember.js* feature allowing us to compute our actual class with help of the attribute *status*. In line 11 we use *Ember.js'* computed-function resulting in an observing of *emotivWebsocket.status* and changing our value *status* in case of *emotivWebsocket.status'* changes. Thereby every change of *status* in Listing 6.2 changes our *status*, too. *Status* is an attribute defining our button's CSS-class and thereby we change the button's color or trigger animations by changing the button's *status*. Furthermore by clicking our button we trigger the start function of our service and connect or reconnect our WebSocket connection.

[2] `https://github.com/thoov/ember-websockets`, as June 2017

Listing 6.2. code parts from our emberservice

```
export default Ember.Service.extend({
  websockets : Ember.inject.service("websockets"),
  socketReference: null,
  status: null,

  start: function(){
    this.set("status", "connecting");
    let sockets = this.get("socketReference");
    sockets = this.get("websockets").socketFor("ws://localhost:1313/");
    sockets.on("open", this.openMessageHandler, this);
    sockets.on("message", this.myMessageHandler, this);
    sockets.on("close", this.myCloseHandler, this);
    this.set("socketReference", sockets);
  },

  openMessageHandler: function(event){
    this.set("status", "connected");
  },

  myMessageHandler: function(event){
  //discussed later
  },

  myCloseHandler: function(event){
    Ember.run.later(this, function(){this.set("status", "disconnected")}, 1000);
  },
});
```

Finally, we can describe our button as some kind of element wrapping our service. We could have written the service's functionalities into the button, but there is a fundamental difference between an *Ember.js* component and an *Ember.js* service. The service is an element outlasting during our application's lifetime, a singleton-object and additionally accessible from all elements in our application by injection, whereas our button as *Ember.js* component does not provide these features. We uphold our service implementation as the safest way to manage our WebSocket connection.

Listing 6.3. Our BCI's Ember.js button

```
export default Ember.Component.extend({
  emotivWebsocket: Ember.inject.service("explorviz-frontend-plugin-bci"),
  classNameBindings: ["status"],
  status : "",
  click: function(){
    this.get("emotivWebsocket").start();
  },

  init(){
    this._super(...arguments);
    this.status=Ember.computed("emotivWebsocket.status", function(){
      return this.get("emotivWebsocket").status;}.bind(this));
  }
});
```

Listing 6.4. Adding an event listener with JQuery

```
$(document)[0].addEventListener("keyup", function(evt){
  if(evt.ctrlKey && evt.key=="b"){
  let controllerID = evt.controller;
  let controllerFlag = evt.controllerFlag;
  this.get("emotivWebsocket").toogleActivation(controllerID, controllerFlag);
  }
}.bind(this));
```

BCI-Mode

The implementation of our BCI-mode matches our design with minor adaption influenced by the VR collaboration. An event listener added on our document with help of *JQuery*[3], a JavaScript library, shown in Listing 6.4 triggers its function in a case of letting go the key sequence ctrl and b. We choose this combination, because it is no shortcut used by browsers. The variables *controllerFlag* and *controllerID* are used to integrate the controller we use in VR as additional tool for controlling our BCI-mode. Our function *toogleActivation* shown in Listing 6.5 changes *status* in our WebSocket service and hinders our *myMessageHandler* in Listing 6.2 from processing received data.

Finally, we want to discuss the changes relating to our considered design of our BCI-mode. The collaboration with VR showed two newly emerging aspects changing our requirements. Firstly, we have to consider performance problems and, secondly, the

[3] https://jquery.com/

Listing 6.5. toogle activation of BCI-mode

```
1  toogleActivation: function(controllerID, controllerFlag){
2    this.set("isActivated", !this.get("isActivated"));
3    if(controllerID){
4      this.get('activatedControllers')[controllerID] = controllerFlag;
5    }
6    //stops the sending of the server
7    if(this.get('socketReference')){
8      this.get('socketReference').send(this.get('isActivated'));
9    }
10 }
```

activation of the BCI-mode by clicking ctrl+b is uncomfortable and difficult for a person beeing in VR-mode.

As a result of our cooperation with the VR bachelor thesis, we decided to reduce the communication between our applications as much as possible. The rendering of a 3D landscape proved itself as performance-heavy to such extent that concurrent processes would produce frame slumps. Consequently, the user would suffer lags in his/her visualization. To soften this problem we used our already integrated BCI-mode and the bidirectional characteristic of our WebSocket connection. While the BCI-mode's only purpose in our design was the activation of our BCI's functionality within the application, we added in our implementation phase an additional usage of our WebSocket connection shown in Listing 6.5 in line 8. We send the newly calculated *isActivated* to our browser-external application. Our browser-external part of our application use this flag to stop or start a sending of its processed data.

Another important aspect influencing our implementation's decision is the integration of VR controllers. A user being in VR-mode is not able to trigger the key sequence ctrl+b easily. Therefore, we give the user the opportunity to activate the BCI-mode by holding a certain button of the controller. Afterwards the user can let got of the button and deactivate the BCI-mode. As well holding as loosing the button simulates a ctrl+b key-up event on the document, which can be fetched by our event listener implemented in Listing 6.4. This abstraction allows us to use our already implemented functions, but we have to write the code for simulating the events in the controller-logic of the *VR-plugin*. In addition to simulating a key-up event, we equip the artificially generated events with the controller's id as *controllerID* and a flag as *controllerFlag* shown in line three and four of Listing 6.4. In our Listing 6.5 in line four we finally safe the *controllerFlag* in a JSON object named *activatedControllers* with the *controllerID* as key. These two attributes are safeguards against the case that the user clicks the certain button on both controllers, but they have another purpose further discussed in Section 6.2.3.

Listing 6.6. Our function for triggering Mental Commands

```
1 triggerMentalCommand(mentalCommand){
2   let canvas = $("#threeCanvas")[0];
3   let event = new Event("bciaction", {"bubbles":true, "cancelable":false});
4   event.mentalCommand = mentalCommand;
5   for(var key in this.get('activatedControllers')){
6     if(this.get('activatedControllers')[key]){
7       event.controllerID = key;
8     }
9   }
10  canvas.dispatchEvent(event);
11 }
```

Changing the Landscape

The last important aspect to understand our plugin is to understand, how we are able to influence our 2D landscape as well as our 3D visualization. In our design we mentioned the problems we see in influencing the *ExplorViz* landscape by our BCI plugin. We suggested a service connecting the relevant *Ember.js* objects. We dismissed this approach, because we found no realistic implementation. Our implemented concept does not use *Ember.js* functionalities in particular, but a concept provided by JavaScript.

The concept is described in Listing 6.6. Independently of the amount of dimensions, we use a canvas with the id *threeCanvas* as platform for our visualization. The *ExplorViz* main application as well as the *VR-plugin* use a canvas with this condition and thereby we fetch this canvas by using a *JQuery* selector in line 2. Afterwards, we trigger a self written event on this canvas with the current mental command sent by our browser-external application and a *controllerID* as additional attributes. The mental commands will always be part of our event and specify the kind of functionality we want to trigger. The *controllerID* is an optional attribute, which only occurs in the *VR-plugin*. We can have two different controllers and so we use our JSON object *activatedControllers* to ask for the id of the controller triggering our BCI-mode. We discuss the actual importance of the id further in Section 6.2.3.

6.2.3 Implementation of our Mental Commands

We discussed, how our plugin communicates with our browser-external application, how our BCI-mode works and how we trigger the functionalities of our mental commands, but we have not yet discussed, what kind of functionalities these are. Primarily, we experimented in the three-dimensional room as result of our collaboration with the VR bachelor thesis and we made some observations influencing our decision, while implementing our design for the mental commands.

Firstly, between our two selected mental commands we were able to evoke the relaxing state easily, but had many problems to concentrate. We came to the conclusion that the functionalities of our mental commands should not be related to each other. In the beginning we tried to use oppositional functionalities for our mental commands, but our incapability of triggering both mental commands with the same ease produced much stress. For example, a user becomes impatient, if he/she is able to lift and object, but not able to drop it. Furthermore the difficulty to reproduce the object's origin state is a problem to such extent that our integration of our device can decrease *ExplorViz'* usefulness enormously.

Secondly, we observed that the additional benefits, which we gain by using a functionality in our visualization, depend on the amount of dimensions. For example, while a changing of the camera can be very useful in the context of a 2D visualization, it produces a sick feeling in a 3D visualization. As consequence we need different functionalities depending on the way of visualization and, thereby, we let the *ExplorViz* main application and the *VR-plugin* decide, what shall happen, while our BCI event occurs. This observation allows us to change the requirements we set for our 2D and 3D visualization.

Our observations open a new opportunity to implement our BCI into the navigation of the virtual room. We know that we can set requirements uncoupled to our 2D visualization and we know that a user interacts with the virtual room with help of two controllers. We already use these controllers to start the BCI-mode, but, additionally, we let the controllers influence the functionality a mental command triggers. For example we want one controller to trigger a lifting and the other to trigger a lowering of our landscape with the same mental command. Therefore we defined a left and a right controller differenced by the color the line emerging from the controllers. This explains, why we always work with a *controllerID* and a *controllerFlag*. We have to differ, which controller triggers the BCI-mode at the moment.

In our evaluation we want to use our mental command triggered by concentration to lift our landscape up and down. We see much use in this function, because the change of perspective allows as user to adapt the landscape's height to her/his benefit. Sometimes the view from further above helps a person to receive a better overview about the landscape structure, while a perspective closer to the object permits a better understanding of details. As the second function we choose a clockwise and a anticlockwise rotation triggered by relaxation. Our VR space is endless and only limited by our computer's resources, but the room of our user's room of movement is often very limited. By rotating the VR landscape a user could also adapt the landscape to his/her interest.

Chapter 7

Evaluation

Due to the newness of our device we have not only to question the mere practicalness but also correlations to aspects regarding the individual personality of users. The BCI reacts to ones mind and, thereby, the insides of our personality influence the effective usability. It is possible that persons, who are better in concentrating, are more suitable to navigate this device. Furthermore, the present state of a mind can influence our capability of handling this device, too. For example an indisposition related to the environment can have influence on the ability to concentrate. In these aspects we see a danger to prove our implementation by a quantitative evaluation from from the start.

We want to focus on the aspect of the user acceptance. Only the user's acceptance will decide, if the device can also take part in our daily life routines. Although the device's accuracy in identifying mental commands is a very important factor, the acceptance is of higher significance. A person accepting the device is more willed to train with it and as consequence the training could counterbalance a low accuracy or a naturally bad ability in handling a BCI. We realized in our implementation some concepts, which can counterbalance a low accuracy and we want to check the users acceptance in relation to these and the user's evaluation regarding his/her control over the device. Our research question is thereby "*Will a user accept our implementation in the context of VR?*". We use following hypothesis to answer our research question.

h1: The user accepts the integration of the BCI in VR.

h2: The user accepts the controller as additional tool.

h3: The user is able to trigger at least one mental command to his/her satisfaction.

7.1 Method

To answer our hypothesis, we chose a qualitative study as our method and performed a collaborated study with the VR bachelor thesis. As additional tool we chose three different surveys. One survey dealt about general questions, one dealt about the VR part and one about our BCI part. The surveys are written in the German language, but for this work we translate the questions into English. Additionally, we recorded the browser interface to have as much valuable information as possible. Furthermore, we use the VR controllers in our study. We marked one controller as left controller and one as right. From the left controller emerges a black ray in the direction the user points, from the right emerges a green one.

7.1.1 The Landscape

In our virtual room we will represent the *ExplorViz* demo-landscape[1] provided in the life demo of *ExplorViz*. All systems of this visualization are closed and, thereby, the user only see gray chest. The landscape is the object the user interacts with in our study .

7.1.2 Mental Commands

The user shall work in our study with two thoughts as mental commands. We use our's implementation's concept. As concentration mental command the user shall take advantage of a ray that emerges out of the controllers in the the virtual room. He/she shall imagine this ray as solid object and try to push some object in the room. Furthermore, we advise the user to move his/her hand in the way he would do in a real process of pushing. As functionalities we choose a lifting of the landscape by holding the right controller, while the left lowers the landscape.

As second mental command we choose relaxation. The user shall try to relax him-/herself while breathing deeply in through nose and breathing deeply and slowly out through the mouth. Holding the button of the left controller produces an anticlockwise rotation, while the right indicates a clockwise rotation.

7.1.3 Choose of Participants

Our BCI group as well as the VR group are interested in seeing, if their application is accepted by the user. For this purpose we reduced our tasks on basic geometric tasks and so the test persons need no basic knowledge related to *ExplorViz*. Thereby, we had not to isolate our group of potential test subjects and so we hoped to gain a big diversity of informations, why our implementation is accepted or not.

7.1.4 Design of our Survey

We ask our participants' evaluation in form of a check questionnaire with the answer possibilities *I agree completely*, *I agree*, *neutral*, *I disagree* and *I disagree completely*. The structure of the BCI survey is inspired by the hypothesis we want to prove. Furthermore, the user has the possibilty to make notes and can note additional improvement proposals.

[1]http://samoa.informatik.uni-kiel.de:8181/login.html;JSESSIONID=a98e6706-2e54-480f-92e1-5dfa787c85c4

h1: The User's Evaluation in Terms of BCI and VR

To prove our hypothesis h1 we have to gain informations about the user's experiences in terms of the BCI and VR. Therefore, we ask the user, if the wearing of BCI and VR is comfortable. This question is important for our hypothesis h1, because the comfort can influence the user's opinion of our VR and BCI concept. Furthermore, we question our participant, if he/she would use our BCI as well as our BCI in combination with the VR again. Thereby we want to receive a direct answer about our hypothesis. We see these three questions in one context to draw conclusion about the participant's answers.

h2: The User's Evaluation in Terms of the Controller as additional Tool

The usage of the controllers is a concept we use to start the BCI-mode and to integrate two functionalities with only one mental command. Additionally, we use the controllers in our study as possibility to strengthen the user's imagination and, thereby, to ease the evocation of mental commands. To test our concept we ask the user, if the usage of the controllers was disturbing or if they helped the user in triggering the concentration mental command. We ask, if the idea of lowering and lifting the landscape with only one mental command is an adequate concept.

h3: The User's Evaluation in Terms of the Mental Commands

We want to know, if the user is able to trigger one mental command to his/her satisfaction. Thereby we ask the participant directly, if he/she was able to lift as well lower the landscape easily and if he/she was able to rotate the landscape easily. Furthermore, we ask the participant further in two different questions, if he/she evaluated the combination between our mental commands and their functionalities as good. Finally, we want to ask the participant, if the device was able to discriminate between the mental commands. With all these question we receive a perception, why or why not he/she wass able to trigger one mental command to his/her satisfaction.

Factors of Disturbance

The human mind is no closed system and thereby manipulable by its environment. Consequently, many factors can influence the result of our evaluation. We use some questions to receive an impression about potential influences and use these results to draw conclusions in the case of the appearance of unexpected results.

We ask, if our participant has long or voluminous hair, because this could have influenced the time we needed for the setup of the gear and, consequently, the test person could have evaluated our product worse. Additionally, the hair could have influenced the contacts between the sensors and the scalp and, thereby, reduce the device's accuracy. Furthermore, we questions our participant, if the time for the training was adequate. This could also influence the user's opinion of our system.

At last, we are interested in our participant's mental state, so we want to know, if our participant felt they were being put under pressure in the study. Additionally, we want to know, if they had the impression they could concentrate themselves. At last we want to know, if our participants felt themselves mentally fit. We regard these question with priority, because this could effect our results to a large extent.

7.1.5 Operational Structure

At first the participant reads an information sheet and will thereby receive some knowledge about the technology we use. Furthermore, he/she will be informed that we will record the display of our computer's monitor, but he will not be recorded. Following, we will plead the participant to fill out the general survey and, afterwards, the VR part of our study will begin. After the participant will have finished the VR part, we will start with our BCI part. In the beginning we will inform the user with a superficial explanation about the BCI as device and the further steps of our study.

Thereafter, the user will put on the VR glasses with our BCI with our help. The next step is the directive training. The user will train three different mental states three times each 8 seconds. In this part the user will have as much time as he needs to prepare him-/herself between the training sessions. The user will dictate us, when we start the training. After a successful training, we will provide the user with the opportunity to repeat this training session or to begin the next one.

Afterwards, we will introduce the user in the BCI-mode and the controllers' functionalities. Subsequent to this introduction the user will get 4 simple tasks. He can solve each two of these tasks by using the concentration mental command and two by using the relaxation mental command. In the meantime we will observe in the console of our browser, how often he/she will evoke the right mental command or the wrong mental command. At last, we will plead the user to answer our BCI survey. In the meantime we will make notes about the procedure.

7.2 Execution

The first half of our study was executed as the VR part. In this part our test persons only wore the VR glasses and made use of the VR controller. They received a tutorial and had to close up with some tasks being reduced on geometrical questions. Afterwards, the test persons answered a survey dealing about their acceptance related to the realization of the virtual room. We saw this first part as a good opportunity to let the user get familiar with the environment, where they will perform our tasks.

Subsequent to the first part of the study we started our part. In the beginning we provided the test person with some basic knowledge about the device. We explained simply that the device measures electricity on the scalp and can thereby identify learned thoughts. After the small introduction into the device and the further processes defining our study, we equipped our test subject with the BCI and the VR glasses and started a directed training session. We trained two different mental commands and a neutral mental state, which acts as a reference for the actual mental commands. Every mental command had three training sessions, which is inspired by the work of Taylor and Schmidt, who indicate an increasing of the device's accuracy after training three times [Taylor and Schmidt 2012]. This is further discussed in Chapter 4.

For our neutral mental state the test person had to think nothing special and should be neither concentrated nor relaxed. After three successful training sessions the user received two VR controller. A ray emerges out of every controller in our virtual room, so we use the rays' colors to identify, which controller has to be held in which hand. Afterwards the test person should think of the ray as some kind of solid object like a pool cue and try to push an object. In the final training stage the test person should try to relax by inhaling deep and slow through the nose and exhaling deep and slow through the mouth.

In connection to our training we showed the user the significance of the BCI-mode and how activating the BCI-mode with the right controller differs with an activation of the left controller. By holding the button of the right controller in our scenario the concentrating mental command triggers a lifting of the landscape and our relaxation mental command triggered a clockwise rotation. By holding the button of the left our concentration mental command triggers a lowering of the landscape and our relaxed state causes an anticlockwise rotation. Furthermore we used this phase to calibrate the sensitivity, which defines, how fast we trigger a mental command.

Followed by the introduction of the controllers' use the test person should rotate the landscape 180 degrees. After this the test person should rotate the landscape in its origin and finally lower the landscape to its origin. In these processes we noted, how often the person successfully triggered the right and how often he/she evoked the wrong mental command. In the end we pleased our test person to answer a survey.

7.3 Results and Discussion

12 persons participated in our study, but, unfortunately, we had to ignore the results relating to our BCI part of two persons. In the end one sensor tip of our device broke and, consequently, the persons were not able to produce any result at all. Consequently, we had only ten participants. Eight of these had no experience with the BCI at all, one person had some experience and another participant had much experience shown in Figure 7.1.

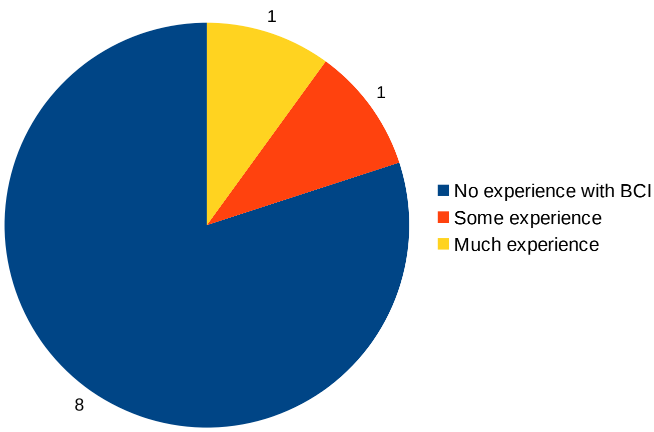

Figure 7.1. Diagram showing the user experience regarding BCI

7.3.1 Results

Our primary interest was in seeing the user's evaluation relating to our implementation, so we had no special interest in evaluating the training. The users had to evaluate their impression regarding the usability of our concepts. Therefore the user had the opportunity to answer some questions. Following we will use some kind of mapping for his/her answer possibilities shown in Table 7.1. Furthermore, we want to show our results in respect to our design of the survey.

Table 7.1. Mapping for evaluation purposes

answer	degree of agreement
I agree completely	4
I agree	3
neutral	2
I disagree	1
I disagree completely	0

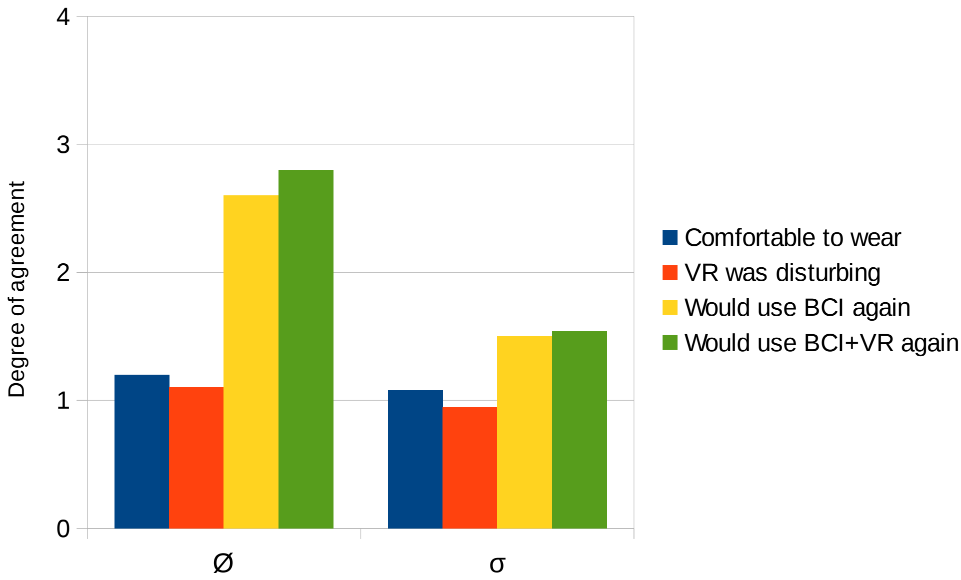

Figure 7.2. Diagram visualizing our results in terms of h1

h1: The User's Evaluation in Terms of BCI and VR

At first, we were interested to see, if they accept the BCI as tool and especially in the context of VR, or if some aspects could have lowered their valuation. Figure 7.2 visualizes our results. Our participants averagely experienced the wearing of the BCI in combination with the VR glasses as not very comfortable. Only two participants evaluated our as construction quite comfortable. Further, the virtual room did not seem to disturb most of our users. Only one participant evaluated the virtual room as disturbing. Our participants would averagely use the BCI as well as the BCI in combination with VR again, but one test person had no interest at all. The standard deviation of all question is close to 1 and thereby we can see that most of our users have a similar opinion relating to our questions.

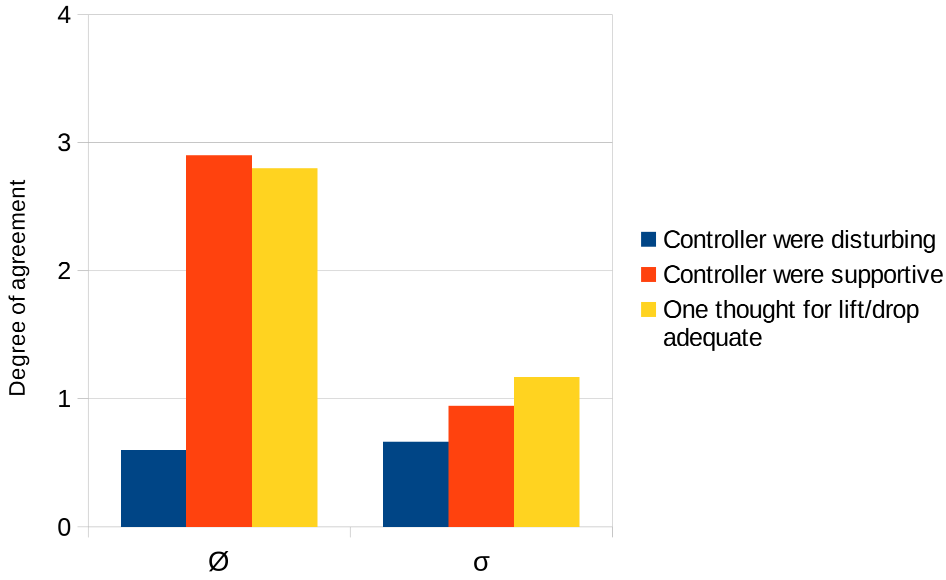

Figure 7.3. Diagram visualizing our results in terms of h2

h2: The User's Evaluation in Terms of the Controller as additional Tool

In the next part we want to discuss the survey's results regarding the usage of the controller shown in Figure 7.3. We can see by the average value in combination with the standard deviation that most of our users did not find the controller disturbing. On the other hand most of our participants agreed in the supportive function of our controllers to evoke the concentration mental command. Only one person said that the controllers have a slightly hindering effect. He/she mentioned that he/she sometimes highlighted some objects in the virtual room, what results in his/her loss of relaxation and thereby he/she was hindered in his/her opinion in evoking our relaxation mental command. Further, the usage of one mental command triggering two oppositional functionalities with only one mental command was averagely evaluated as adequate.

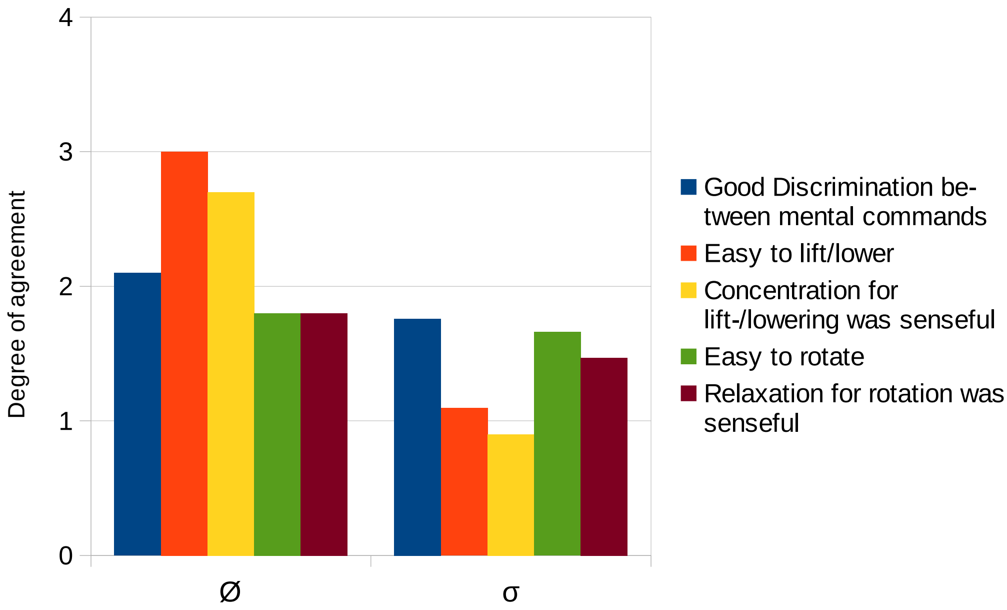

Figure 7.4. Diagram visualizing our results in terms of h3

h3: The User's Evaluation in Terms of the Mental Commands

Figure 7.4 shows our results regarding the user's own estimation in terms of the mental commands. We achieved no concordant answer in the question, if the device appropriately differs both mental commands. We see a very big difference in the opinion of our test persons in form of the standard deviation. While four of our participants completely agreed, three disagreed. In this context our average value loses its significance.

Averagely, our users had the feeling that they could lift the landscape well. Only two users seem to have problems in their opinions. According to the notes we made in their cases both of the users were able to lift the landscape, but showed problems in lowering the landscape afterwards. Further, the combination between concentration and lifting the landscape was averagely meaningful in the eyes of our test persons and our low standard deviation shows no big diversity in the participant's evaluation.

The participants' evaluation regarding the rotation shows a very high diversity. Our average values are just slightly bigger than our standard deviations. This is the result of a high diversity of answer regarding this question.

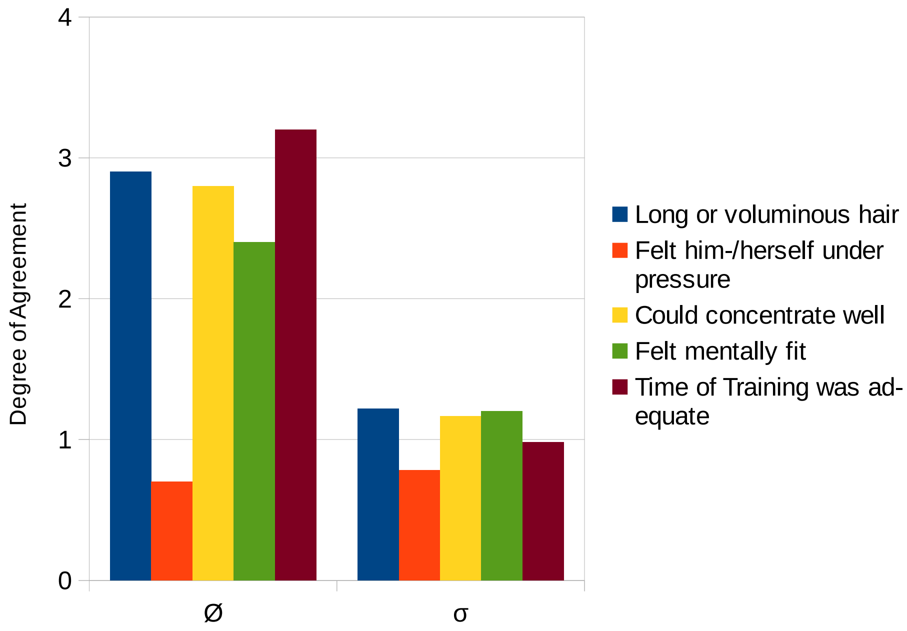

Figure 7.5. Diagram showing our results in terms of factors of disturbance

Factors of Disturbance

Our results regarding our factors of disturbance are shown in Figure 7.5. At first, we notice that the standard deviation of each question was near 1 and, consequently, we can reduce our results on the average values. Most of our participants had long or voluminous hair in their opinion. Furthermore, they did not feel pressure averagely in the study. The participants' ability to concentrate was averagely slightly good, while they described their mental fitness neutral. Averagely, the users experienced the time for training as adequate. Only one person thought of the training time as not adequate. We noted, that the BCI did not fit the person well and we had to repeat more training sessions than usual.

7.3.2 Discussion

We want now to discuss our hypotheses to answer our research question.

h1: The User's Evaluation in Terms of BCI and VR

As already mentioned, many aspects could influence the user's evaluation regarding our implementation For example the head's form had much influence of the BCI's fitting in combination with the VR glasses. Additionally, the wearing of own glasses and the volume of hair had much effect on the time needed for the setup. We often had to clear the contact between the scalp and the sensors. Further, we observed pressure points on the places, where the sensors had contact to the scalp. Our Observations in relation to the bad siting of the BCI in combination with the VR gear found support in the answers of our survey. Most of our subjects described the wearing as uncomfortable. As consequence we expected a bad valuation of the usage of our BCI, but contrary to our expectations most of our test subjects would use the BCI again and rather the BCI in combination with the VR. Some participants even uttered their entertainment in using the device, but they also mentioned that they would not use this device for too long. We see this contradiction as a sign for a high acceptance for the BCI as additional tool in VR and thereby verify our hypothesis h1.

h2: The User's Evaluation in Terms of the Controller as additional Tool

Additionally, the integration of the VR controller proved itself as accepted concept. Only one participant was at least slightly disturbed, but most had no problem with the usage at all and even liked the supportive function of our controllers. Most of our test subjects even were not disturbed to use the same mental command for lifting and dropping the landscape. Furthermore, one test subject noted that he/she did not see much additional benefit in using the mental command while holding the button. Additionally, two test persons criticized that they did not hold the button in their training and this could have some effect. Fairly, these are valid criticisms against our process of the training, but it is no criticism against the usage of the controller. Finally, we would evaluate the integration of the controller as rather a additional benefit than a hindrance and thereby verify our hypothesis h2.

h3: The User's Evaluation in Terms of the Mental Commands

Most of our participants consider themselves as capable or very capable in triggering the concentrated mental command, but two persons had much problems in dropping the landscape afterwards. On the other hand these two persons had less problems lifting the landscape. The participants evaluated our relaxation mental command as worse than the other mental command. They valued averagely our combination between mental command and rotation as neutral or bad. Additionally, we gained much feedback that they loose their relaxation, when the landscape starts rotating. Furthermore, while the first half of our

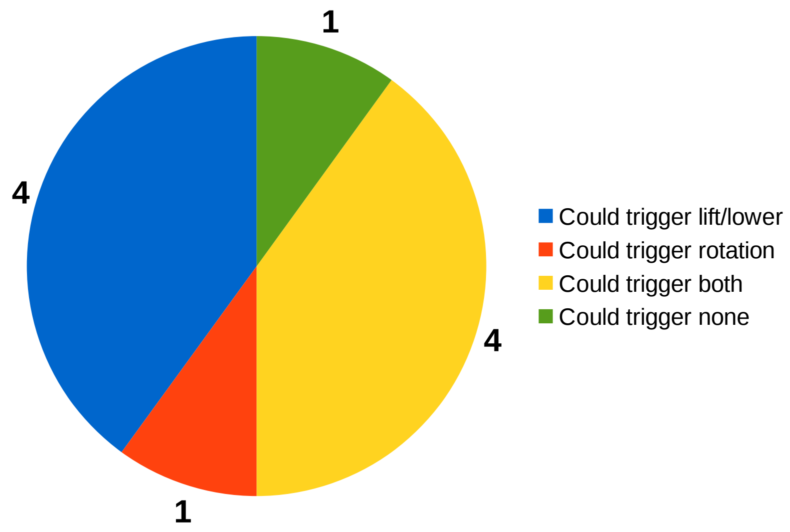

Figure 7.6. Diagram representing our answer for h3

participants described its evocation as easy, the other half described it as difficult. Although our notes support their impression, we observed additionally a commonality between our participants. In most of the cases we noticed a delay between the start of relaxation and the start of rotation. We expect that a person needs some time to reach a relaxed state, but this delay can also produce dissatisfaction and consequently stress. Indeed, we think that relaxation was the wrong approach, but we see a commonality between the persons, who were able to evoke the rotation perfectly. They all concentrated on their breathing technique. One person even used some yoga techniques with strong inhalation and exhalation and two other mentioned their incapacity of triggering the rotation without consciously breathing. Maybe we should not have focused on relaxation but only on inhalation. It is to mention that two of the three persons being able to evoke the relaxation perfectly, also triggered the concentration perfectly.

We finally want to answer our hypothesis by the usage of Figure 7.6. We categorized a person in the group *Could trigger lift/lower*, if he/she agreed(3) or completely agreed(4) in being able to lift/lower the landscape easily. The other groups are categorized by the same concept. We see, that nine of ten persons were able to trigger at least one mental command to their satisfaction and, thereby, we see our hypothesis as limitedly verified.

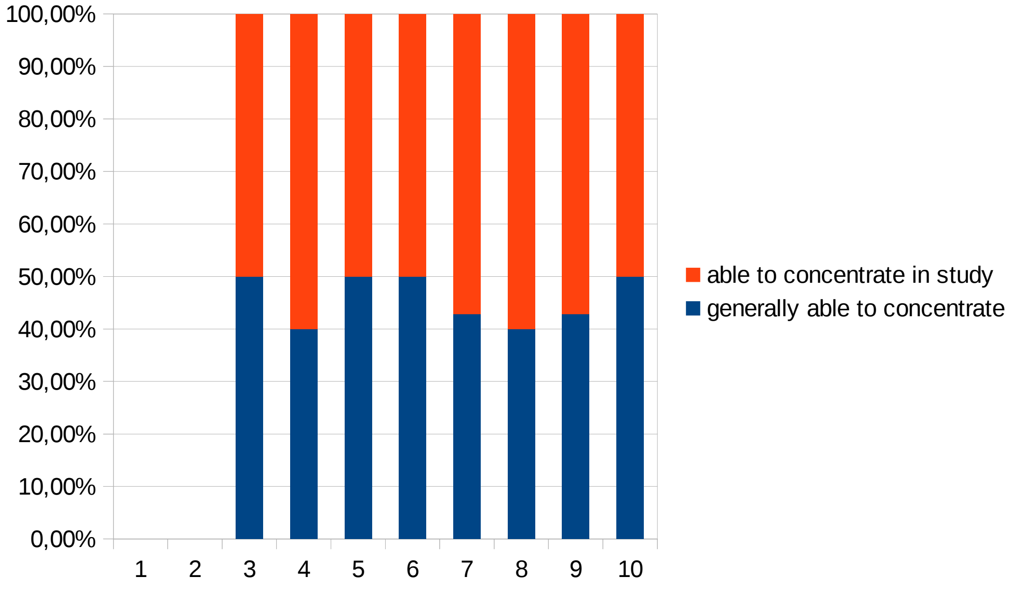

Figure 7.7. Diagram showing the relation of their general ability to concentrate and their ability to concentrate in our study

Factors of Disturbance

We did not find any connection between our factors of disturbance and a possible danger for our hypothesis. The long hair did not clearly influenced the person's ability to trigger his/her mental commands. The training time did not clearly influenced the person's will to use VR and BCI again and a bad concentration did not correlate with the ability to control the device.

On the other hand we noticed an interesting correlation with the help of our general survey. In the general survey we asked our participant, if he/she is generally good in concentrating and in our BCI survey we asked our test subject, if he/she was able to concentrate well. In Figure 7.7 we see that every participant evaluated his/hers ability to concentrate in our study higher or equal than his/hers general ability of concentration. This could be a coincidence, but at least we want to note that this could also support our hypothesis. If the user regards him-/herself better in concentrating in the virtual room than he/she usually is, this would indicate that the controller helped him/her concentrating or that the virtual room had a positive effect on his/hers concentration.

Conclusion of our Discussion

We were able to verify all of our hypotheses and, thereby, we see the BCI as a possibly additional tool in the context of VR, but to test our products actual usability more studies are needed. We need much more evaluations to receive an impression, if our implementation is suitable for complex real task comprehensions.

In this project we produced informations to such extent that we were just limitedly able to evaluate all appropriately regarding the human computer interaction. This results out of the different personalities of our users. For example, we gained the impression that introverted person were better able handle the device. These aspects can only be probed with the work of quantitative studies.

7.4 Threats of Validity

It is difficult to name all possible threats for a evaluation dealing about ones mind, but we want to name our main expectations. At first, we see a high diversity in our own performance in the evaluation. We had to explain people, what they had to think and this was more individual than expected. Some people understood something different in our instructions and, often, we had to correct some behavior. For example many persons did not inhaled deeply and slowly, until we added that they need to hear their breathing. These variations made every training session different. Furthermore, we had to often adapt our working processes in the study. If a person was not able to rotate the landscape, we had to adapt to this and had to change further instructions.

Additionally, we can not eliminate the suspicion that the own valuation of the combination between mental command and its function has much influence on the capability of evoking the function. Another approach would be to let the participants decide, what function shall be triggered with their mental command. On the other hand this individualization would influence our results in a study.

Another important aspect is the delay between the start of relaxation and the start of influencing the landscape. This delay was unexpected, but can also be normal. This could have effect on our test person to such extent that he/she could not relax anymore. We could have prevent this circumstance by informing and preparing the test person. A further possible threat is the collaborated study. Our participant could be tired after the VR part or the circumstance that he/she had two different study conductors could also have an effect on the participants mind.

The last and most important aspect could have been the damaged sensor tips. We can not eliminate the possibility that the sensor was defect from the beginning. Considering this, it is possible that our mental commands were not recognized properly and this would have influenced our study definitely.

Chapter 8

Conclusion

Our integration showed itself as problem-free. Our concept of the WebSocket connection was not only the safest approach for integrating the BCI into *ExplorViz* but also probably the only possible enabling us to integrate the device with the *VR-plugin*. The VR renderer needs computer-based resources to such extent that we can not process more data concurrently. If we had chosen a complete browser-internal application, we were never able to make the collaboration study with the VR. Hence, we are not sure, if a complete browser-internal solution would be general good idea. Furthermore, our browser-internal concept of using events for communicating with *ExplorViz* proved itself as flexible enough to even communicate with the *VR-plugin*. Finally, we want to discuss our browser-external approach. Our model's structure based on the decorator pattern made it easy to change the program flow. We experimented a lot and were able to exchange decorator objects and thereby change the program parts without much consideration of the complex event handling. Furthermore we had a good control over the thread, who let the model actualize itself. On the other hand we have to admit that we neglected the user interface. A user would probably have problems using the GUI.

In terms of our evaluation we were able to prove our hypothesis and, consequently, see the BCI as additional tool in VR. Our participants accepted the BCI as tool in our VR room and accepted the controller as additional tool to interact with BCI and VR. Further, almost every participant was able to trigger at least one mental command to her/his satisfaction. By following our research question we received much insight into the complexity of the use of a BCI. Some of our participants liked our combination between our relaxation and the rotation as functionality. We are not able to say, if they would have been better in triggering another functionality by relaxation. We are also not able, if breathing would be a better approach as mental command. Personally, we were good in triggering the rotation, but maybe we also worked unconsciously more with the breathing than with relaxation. Afterthought, we think that the short-term study was probably not the right approach for testing our implementation. To many aspects regarding own preference, the current mental state and the person's individual characteristics make it difficult to test all persons under the same conditions with expecting best possible results. We remain with our opinion that the usage of one mental command after 72 seconds training is an appropriate result regarding the fact that almost no participant had any experience with the BCI at all and we have found many interesting aspects in terms of the relationship between this new kind of interface and the human being.

We would summarize our results as followed: In the process of implementing the BCI as additional tool in *ExplorViz* we received many informations regarding this new device and we were able to build a highly adaptable software. The browser-external part of our software enables a fast changing of our application's functionality without needing any informations about the device's event management or deep insight into the nature of brain activity. The concept of our WebSocket connection provides the opportunity to integrate our browser-external part in almost all Web applications in a few hours. Thereby the device's functionality could be integrated in other applications for task comprehension. Our browser-internal part's structure enabled us to integrate *Emotiv Insight's* functionality in the *VR-plugin* in only approximating four hours. Consequently, we can also implement the *Emotiv Insight's* functionality in other visualizations *ExplorViz* could possibly provide in the future. In our evaluation we received much insight into the complexity of the relationship between human and interface, but we explored the BCI as a possibility for new task comprehensions in terms of user acceptance. The modularity of our software and the realization make this work for a proper foundation to explore more new paths of task comprehensions.

Chapter 9

Future Work

Our work has much potential for future work. We see the two most important aspects in the fields of individualization of our mental commands and the GUI. We recognized in our evaluation that the opinions about the combination between mental command and its sfunctionality are very different, so we believe that the best solution would be in letting the user decide, what his/her mental command shall trigger. We would propose for this purpose to implement into our plugin some configuration possibilities.

Further we see a development of our GUI as important. we neglected this aspect and, thereby, our application is not suitable for users. At the moment the user has only the possibility to train the neutral state and two mental commands, but the device provide more features like recognizing facial expression and a profile management. Depending on the recognition of facial expression this could also offer new ways of task comprehensions. In our evaluation we also missed some ways to see the signals' strength. The possibility of receiving information about the contact quality is integrated in the device categorically, but this function did not work, when we started with our integration. Meanwhile we have updated the libraries, but we have never tried to implement this feature again.

For the time being we would also advise against a complete browser-internal solution for our plugin. Thereby we fear about destroying the connectivity to the *VR-plugin*, because our plugin and the VR render would probably fight about the browser resources. This work should wait for new browser concepts or a increased render-performance.

We see a further interesting possibility of our WebSocket connection. Our WebSocket connection is bidirectional and, therefore, we see the opportunity to run our browser-external part in the background, but let the user control the browser-external part by sending commands over the WebSocket connection. Thereby, we maintain the performance of our software, its modularity and are able to provide possibilities to train his mental commands by only using the browser. In the context of VR we could even integrate some kind of user interface to let him control our browser external part.

Finally, we would also advise to use for further evaluations the *Emotiv Epoch*[1]. This BCI is another BCI from *Emotiv Systems* and provides more 14 EEG sensors and 2 EEG sensors and is thereby probably more accurate in differing mental commands. This device looks more flexible, consequently more comfortable and more suitable for long-term studies and further developing processes. We also expect a better accuracy enabling a better usability.

[1] https://www.emotiv.com/epoc/

Bibliography

[Babiloni et al. 2004] F. Babiloni, C. Babiloni, F. Carducci, G. L. Romani, P. M. Rossini, L. M. Angelone, and F. Cincotti. Multimodal integration of EEG and MEG data: A simulation study with variable signal-to-noise ratio and number of sensors. *Human brain mapping* 22.1 (2004), pages 52–62. (Cited on page 7)

[Burmaoglu 2002] E. Burmaoglu. Brain Power - Gehirn steuert Computer (2002). (Cited on pages 5 and 7)

[Cravens and Brady 2014] J. Cravens and T. Q. Brady. *Building Web Apps with Ember. js: Write Ambitious Javascript*. " O'Reilly Media, Inc.", 2014. (Cited on page 12)

[Fette 2011] I. Fette. The websocket protocol (2011). (Cited on page 10)

[Fichter 2002] C. Fichter. Brain-Computer Interfaces (2002). (Cited on page 5)

[Fittkau et al. 2017] F. Fittkau, A. Krause, and W. Hasselbring. Software landscape and application visualization for system comprehension with ExplorViz. *Information and software technology* 87 (2017), pages 259–277. (Cited on page 12)

[Fittkau et al. 2015] F. Fittkau, S. Roth, and W. Hasselbring. ExplorViz: Visual runtime behavior analysis of enterprise application landscapes. In: AIS. 2015. (Cited on pages 1 and 12)

[Fittkau et al. 2013] F. Fittkau, J. Waller, C. Wulf, and W. Hasselbring. Live trace visualization for comprehending large software landscapes: The ExplorViz approach. In: *Software Visualization (VISSOFT), 2013 First IEEE Working Conference on*. IEEE. 2013, pages 1–4. (Cited on page 12)

[Graimann et al. 2010] B. Graimann, B. Z. Allison, and G. Pfurtscheller. *Brain-computer interfaces: Revolutionizing human-computer interaction*. Springer Science & Business Media, 2010. (Cited on pages 6, 7)

[Kaku 2014] M. Kaku. Die Physik des Bewusstseins. *Über die Zukunft des Geistes, Reinbek* (2014). (Cited on pages 5, 6)

[Lang 2012] M. Lang. Investigating the Emotiv EPOC for cognitive control in limited training time. *Honours report, University of Canterbury* (2012), page 8. (Cited on page 7)

[Levicán et al. no date] C. Levicán, A. Aparicio, V. Belaunde, and R. F. Cádiz. Insight2OSC: using the brain and the body as a musical instrument with the Emotiv Insight (). (Cited on page 15)

[Ochoa 2002] J. B. Ochoa. EEG signal classification for brain computer interface applications. *Ecole Polytechnique Federale De Lausanne* 7 (2002), pages 1–72. (Cited on pages 5, 6)

[Powell et al. 2013] C. Powell, M. Munetomo, M. Schlueter, and M. Mizukoshi. Towards thought control of next-generation wearable computing devices. In: *International Conference on Brain and Health Informatics*. Springer. 2013, pages 427–438. (Cited on pages 7 and 15)

[Renard et al. 2010] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, and A. Lécuyer. OpenViBE: an open-source software platform to design, test, and use brain-computer interfaces in real and virtual environments. *Presence: teleoperators and virtual environments* 19.1 (2010), pages 35–53. (Cited on page 7)

[Rowland et al. 1985] N. Rowland, M. Meile, S. Nicolaidis, et al. EEG alpha activity reflects attentional demands, and beta activity reflects emotional and cognitive processes. *Science* 228.4700 (1985), pages 750–752. (Cited on page 21)

[Shaker 2006] M. M. Shaker. EEG waves classifier using wavelet transform and Fourier transform. *brain* 2 (2006), page 3. (Cited on page 6)

[Taylor and Schmidt 2012] G. S. Taylor and C. Schmidt. Empirical evaluation of the Emotiv EPOC BCI headset for the detection of mental actions. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Volume 56. 1. SAGE Publications Sage CA: Los Angeles, CA. 2012, pages 193–197. (Cited on pages 15 and 37)

[Vallabhaneni et al. 2005] A. Vallabhaneni, T. Wang, and B. He. "Brain—computer interface". In: *Neural engineering*. Springer, 2005, pages 85–121. (Cited on pages 6, 7)

[Zirkelbach 2017] C. Zirkelbach. Juggling with Data: On the Lack of Database Monitoring in Long-Living Software Systems. In: *4th Collaborative Workshop on Evolution and Maintenance of Long-Living Software Systems (EMLS)*. Feb. 2017. (Cited on page 1)

[Zirkelbach et al. 2015] C. Zirkelbach, W. Hasselbring, and L. Carr. Combining Kieker with Gephi for Performance Analysis and Interactive Trace Visualization. In: *Symposium on Software Performance 2015: Joint Developer and Community Meeting of Descartes/Kieker/Palladio*. Softwaretechnik-Trends, Nov. 2015. (Cited on pages 1 and 12)

Sehr geehrter Proband/sehr geehrte Probandin,

wir bedanken uns im Vorfeld der Studie für Ihre Teilnahme und möchten Ihnen ein paar allgemeine Informationen geben.
Diese Studie ist anonym und es werden keine persönlichen Informationen nach außen geliefert. Uns ist weiterhin wichtig, dass Sie zur Kenntnis nehmen, dass wir Sie bitten unser Produkt zu bewerten. Aus diesem Grund haben wir uns im Vorfeld gegen Laufzeit-Aufnahmen von Ihnen (z.B. Bild und Ton) entschieden. Wir zeichnen lediglich den Bildschirm auf, der keine Verbindung zu Ihrer Person ermöglicht. Wir werden Sie bitten bestimmte Aufgaben zu lösen und uns am Ende einen Fragebogen zu beantworten.

Im Laufe dieser Studie werden Sie mit zwei Technologien in Berührung kommen: Der VR-Brille und dem Brain-Computer-Interface.
Bei der VR-Brille handelt es sich um eine Brille, die dem Nutzer einen 3-dimensionalen Raum aufzeigt, in dem der Nutzer interagieren, sich wie gewohnt umschauen und bewegen kann. Der Fußboden dieses Raums entspricht dem realen Raum, der für Bewegungen zur Verfügung steht und sollte deshalb nicht verlassen werden. In dieser Studie wird Ihnen in diesem Raum eine Darstellungsform einer Software (oder auch Computerprogramm) gezeigt. Wir werden diese als Landschaft bezeichnen. Es ist kein Vorwissen bzgl. eines Programms gefordert. Wir werden lediglich geometrische Fragen stellen. Die dargestellte Landschaft besteht aus Systemen, die als graue Boxen dargestellt sind. Diese Boxen enthalten Teilsysteme, die durch Öffnen dieser Boxen dargestellt werden. Eine geöffnete Box wird als Fläche dargestellt, auf der sich die Teilsysteme befinden. Die Teilsysteme werden als grüne Boxen oder blaue Flächen dargestellt. Aus den blauen Flächen können besondere Teilsysteme erzeugt werden. Diese werden als Applikationen bezeichnet und bestehen ebenfalls aus Boxen und Flächen. Die Interaktion mit den Boxen und Flächen erfolgt durch zwei Controller. Diese verfügen im virtuellen Raum über einen Strahl, der wie ein Laserpointer funktioniert. Mit dem Strahl anvisierte Boxen und Flächen werden rot markiert, wenn mit ihnen interagiert werden kann. Zu jedem anvisierten Objekt der Landschaft oder Applikation kann ein Informationstext angezeigt werden.
Das Brain-Computer-Interface (BCI) ist ein neuro-technisches Stirnband, welches die Spannung auf der Kopfoberfläche misst und darauf reagiert. Dadurch ist das Gerät begrenzt in der Lage bestimmte Gedanken wieder zu erkennen und darauf zu reagieren. Nach einer kurzen Trainingsphase sollen Sie das BCI nutzen, um ein paar simple Aufgaben zu lösen.

Allgemeine Angaben zur Person

ID: ___
Studiengang: ______________________, Fachsemester ____
Lerntyp: ______________________
Lernen durch Schreiben, Lesen etc.
Abschluss : ______________________
Bei bereits abgeschlossenem Studium

	stimme völlig zu	stimme eher zu	neutral	stimme eher nicht zu	stimme gar nicht zu
Ich fühle mich fit und gesund	☐	☐	☐	☐	☐
Ich bin anfällig für Motion Sickness (Reisekrankheit, Seekrankheit, etc.)	☐	☐	☐	☐	☐
Ich habe eine Sehschwäche	☐	☐	☐	☐	☐
Ich trage eine Brille	☐	☐	☐	☐	☐
Ich lerne schnell	☐	☐	☐	☐	☐
Ich werde leicht nervös	☐	☐	☐	☐	☐
Ich bin aufgeregt	☐	☐	☐	☐	☐
Ich kann mich gut konzentrieren	☐	☐	☐	☐	☐
Ich stehe unter dem Einfluss von Koffein	☐	☐	☐	☐	☐
Ich stehe unter dem Einfluss von Nikotin	☐	☐	☐	☐	☐
Ich stehe unter dem Einfluss von Medikamenten	☐	☐	☐	☐	☐
Ich habe bereits Erfahrung mit virtueller Realität gemacht	☐	☐	☐	☐	☐
Ich habe bereits Erfahrung mit Brain-Computer-Interfaces gemacht	☐	☐	☐	☐	☐

Proband	7	8	11	94	37	6	3	29	54	134	Ø	σ
Ich fühle mich fit und gesund	4	4	4	4	3	4	4	4	4	3	3,8	0,4
Ich bin anfällig für Motion Sickness	1	0	2	1	0	1	0	2	0	1	0,8	0,75
Ich habe eine Sehschwäche	3	0	3	4	2	4	3	1	0	3	2,3	1,42
Ich trage eine Brille	3	1	0	4	2	4	4	0	0	4	2,2	1,72
Ich lerne schnell	2	1	2	3	3	2	2	1	4	3	2,3	0,9
Ich werde leicht nervös	4	0	1	3	1	1	1	3	2	3	1,9	1,22
Ich bin aufgeregt	3	1	0	1	0	0	1	3	0	3	1,2	1,25
Ich kann mich gut konzentrieren	0	0	3	2	2	2	3	2	3	3	2	1,1
Ich stehe unter dem Einfluss von Koffein	4	3	4	0	0	4	0	0	0	0	1,5	1,86
Ich stehe unter dem Einfluss von Nikotin	0	0	0	0	0	0	0	0	4	0	0,4	1,2
Ich stehe unter dem Einfluss von Medikamenten	0	0	0	0	0	0	0	0	0	0	0	0
Ich habe bereits Erfahrung mit virtueller Realität gemacht	0	4	3	1	4	4	1	3	0	3	2,3	1,55
Ich habe bereits Erfahrung mit dem Brain-Computer-Interface gemacht	0	0	0	0	1	4	0	0	0	0	0,5	1,2

Virtuelle Realität (VR)

	stimme völlig zu	stimme eher zu	neutral	stimme eher nicht zu	stimme gar nicht zu
Das Erkunden der virtuellen Umgebung war angenehm	☐	☐	☐	☐	☐
Die Darstellung der virtuellen Umgebung war angenehm	☐	☐	☐	☐	☐
Die VR-Brille war komfortabel zu tragen	☐	☐	☐	☐	☐
Das Umschauen im virtuellen Raum durch Drehen des Kopfes oder Drehen des Körpers war intuitiv	☐	☐	☐	☐	☐
Die Annäherung an Objekte durch Daraufzugehen oder durch Annähern des Kopfes war intuitiv	☐	☐	☐	☐	☐
Die Gewinnung von Abstand zu Objekten durch Wegbewegen war intuitiv	☐	☐	☐	☐	☐
Die Fortbewegung im virtuellen Raum durch reales Gehen war praktisch	☐	☐	☐	☐	☐
Die Körper- und Kopfbewegungen zur Navigation (umschauen, annähern, distanzieren) waren leicht zu lernen	☐	☐	☐	☐	☐
Die Interaktion mit der virtuellen Umgebung durch die Controller war intuitiv	☐	☐	☐	☐	☐
Die Bedienung der Controller war leicht zu lernen	☐	☐	☐	☐	☐
Das automatische Markieren von Objekten durch den Strahl des Controllers war praktisch	☐	☐	☐	☐	☐
Durch die Markierung habe ich schnell verstanden mit welchen Objekten interagiert werden kann	☐	☐	☐	☐	☐
Das Öffnen und Schließen von Boxen durch den Abzug war praktisch	☐	☐	☐	☐	☐
Das Rotieren, Verschieben und Zoomen eines Objektes durch Gedrückthalten des großen runden Knopfes war praktisch	☐	☐	☐	☐	☐
Beim Rotieren, Verschieben oder Zoomen eines Objektes hatte ich das Gefühl, dieses in der Hand zu halten	☐	☐	☐	☐	☐
Die Darstellung von Informationen in einem Fenster neben dem Controller war praktisch	☐	☐	☐	☐	☐
Ich konnte dieses Fenster ähnlich wie eine Zeitung in die Hand nehmen und meinem Blickwinkel anpassen	☐	☐	☐	☐	☐
Die Rückmeldung der virtuellen Umgebung war direkt und flüssig	☐	☐	☐	☐	☐
Ich war frei von Schwindel und Übelkeit	☐	☐	☐	☐	☐
Ich hätte noch weitere Zeit in der virtuellen Umgebung verbringen können	☐	☐	☐	☐	☐
Ich würde den VR-Modus wiederverwenden	☐	☐	☐	☐	☐

Anmerkungen:

Verbesserungsvorschläge:

Brain-Computer-Interface (BCI)

	stimme völlig zu	stimme eher zu	neutral	stimme eher nicht zu	stimme gar nicht zu
Ich habe lange oder volle Haare	☐	☐	☐	☐	☐
Das Tragen des BCIs mit der VR-Brille war angenehm	☐	☐	☐	☐	☐
Ich fühlte mich während der Studie unter Druck gesetzt	☐	☐	☐	☐	☐
Ich konnte mich gut konzentrieren	☐	☐	☐	☐	☐
Die Nutzung der Controller war störend	☐	☐	☐	☐	☐
Ich habe mich leistungsfähig gefühlt	☐	☐	☐	☐	☐
Die Zeit zum Trainieren der Gedanken war angemessen	☐	☐	☐	☐	☐
Die virtuelle Umgebung hat mich beim Fassen der Gedanken gestört	☐	☐	☐	☐	☐
Das Gerät konnte präzise zwischen den beiden Gedanken unterscheiden	☐	☐	☐	☐	☐
Ich konnte die Landschaft leicht anheben und absenken	☐	☐	☐	☐	☐
Das Anheben der Landschaft durch Konzentrieren war sinnvoll	☐	☐	☐	☐	☐
Das Rotieren der Landschaft durch Entspannen war sinnvoll	☐	☐	☐	☐	☐
Ich konnte die Landschaft leicht rotieren.	☐	☐	☐	☐	☐
Die Unterstützung der Controller beim Erzeugen des Gedankens zum An- und Abheben der Landschaft war sinnvoll	☐	☐	☐	☐	☐
Für das An- und Abheben der Landschaft war ein einziger Gedanke ausreichend.	☐	☐	☐	☐	☐
Ich würde das BCI wiederverwenden	☐	☐	☐	☐	☐
Ich würde das BCI in Kombination mit virtueller Realität wiederverwenden	☐	☐	☐	☐	☐

Anmerkungen:

____

____

____

Verbesserungsvorschläge:

____

____

____

Notizen zum Ablauf (BCI)

ID: ___

	stimme völlig zu	stimme eher zu	neutral	stimme eher nicht zu	stimme gar nicht zu
Das BCI war einfach anzubringen	☐	☐	☐	☐	☐

Allgemeine Bemerkungen:

Verlauf:

Besonderheiten:

	MC1	MC2
MC1		
MC2		

	MC1	MC2
MC1		
MC2		

		Proband	7	8	11	94	37	6	3	29	54	134	Ø	σ
factors of disturbance		Ich habe lange oder volle Haare	4	2	4	4	2	1	4	1	3	4	2,9	1,22
		Ich fühlte mich während der Studie unter Druck gesetzt	0	0	2	0	1	2	0	1	0	1	0,7	0,78
		Ich konnte mich gut konzentrieren	4	0	3	3	2	2	4	3	4	3	2,8	1,17
		Ich habe mich leistungfähig gefühlt	4	4	2	2	2	2	4	2	4	2	2,8	1
		Die Zeit zum Training der Gedanken war angemessen	4	4	4	3	3	2	4	3	4	1	3,2	1
h1		Das Tragen des BCIs mit der VR Brille war angenehm	3	0	1	1	0	0	2	1	1	3	1,2	1,08
		Die virtuelle Umgebung hat mich beim Fassen der Gedanken gestört	0	0	1	1	2	0	1	3	2	1	1,1	0,94
		Ich würde das BCI wiedervernden	4	1	0	2	4	1	4	4	4	2	2,6	1,5
		Ich würde das BCI mit Kombination der virtuellen Realität wiederverwenden	4	1	0	2	4	1	4	4	4	4	2,8	1,54
h2		Die Nutzung der Controller war störend	0	0	1	1	0	2	0	1	0	1	0,6	0,66
		Die Unterstützung der Controller bei dem An- und Abheben der Landschaft war sinnvoll	4	4	3	4	2	2	3	3	3	1	2,9	0,94
		Für das An- und Abheben der Landschaft war ein Gedanke ausreichend	4	4	3	1	1	2	3	4	2	4	2,8	1,17
h3		Das Gerät konnte präzise zwischen den beiden Gedanken unterscheiden	4	4	0	1	3	0	1	4	4	0	2,1	1,76
		Ich konnte die Landschaft leicht anheben und absenken	4	4	3	3	3	1	1	4	4	3	3	1,1
		Das Anheben der Landschaft war sinnvoll	4	2	3	1	3	3	3	4	2	2	2,7	0,9
		Ich konnte die Landschaft leicht rotieren	4	3	0	0	1	0	4	3	3	0	1,8	1,66
		Das Rotieren der Landschaft durch Entspannung war sinnvoll	4	0	0	0	1	2	4	3	2	2	1,8	1,47
tasks	lifting	right command	9	4	12	13	9	7	5	8	18	13		
		wrong command	0	6	0	0	0	0	3	0	0	0		
		Ø	1	0,4	1	1	1	1	0,63	1	1	1		
	rotation 1	right command	23	13	38	0	21	8	38	22	23	0		
		wrong command	6	3	15	0	5	4	0	22	23	12		
		Ø	0,79	0,81	0,72	-	0,81	0,67	1	0,5	0,5	0		
	lowering	right command	8	5	16	8	9	2	0	22	10	0		
		wrong command	3	0	0	0	0	5	37	0	0	0		
		Ø	0,73	1	1	1	1	0,29	0	1	1	-		
	rotation 2	right command	35	7	0	0	18	3	39	24	17	13		
		wrong command	7	5	47	8	15	0	0	0	0	5		
		Ø	0,83	0,58	0	0	0,55	1	1	1	1	0,72		
	Ø	lift/drop	0,86	0,7	1	1	1	0,64	0,31	1	1	0,72	0,82	0,22
		rotation	0,81	0,7	0,36	0	0,68	0,83	1	0,75	0,75	0,36	0,66	0,28

Notizen zum Ablauf (VR)

ID: ___

	stimme völlig zu	stimme eher zu	neutral	stimme eher nicht zu	stimme gar nicht zu
Die VR-Brille war einfach anzubringen	☐	☐	☐	☐	☐
Der Proband hat das höchste System gefunden	☐	☐	☐	☐	☐

Allgemeine Bemerkungen:

Verlauf:

Besonderheiten: