Abstract

Being fatigued is a common state of everyone's daily life. It is a natural sign of your body that signals you to have a break and rest. However, as soon as this state of fatigue lasts longer and does not go back through short breaks, you could consider yourself being cognitively fatigued. This type of fatigue persists after long-term cognitive and mental stress. The purpose of this report is to look at a methodology to use a cross platform application to determine cognitive fatigue.

Introduction

Cognitive fatigue arises from prolonged periods of time in which mentally demanding tasks are performed that induce a state of subjective exhaustion, feelings of "tiredness and lack of energy", and decline in performance [1]. It can be said that mental fatigue is felt much more often after mentally strenuous tasks than after physical exertion. Examples that can lead to cognitive fatigue are, for example, strenuous work days without appropriate compensation or processing the tax return [2]. Especially with prolonged mental stress, we often feel drained and have the feeling that we can no longer continue. A classic example of this is a long drive. This strains our brain so much that after a certain time we are no longer able to drive and need a break, although the body itself did not have to work hard [2]. Symptoms of cognitive fatigue can mainly be described as follows: poor concentration, forgetfulness, slight irritation, simple tasks seem difficult to solve, feeling overwhelmed, physical stress symptoms such as headaches, tension and an increased heart rate, feeling of a mental block, you feel foggy and sleep is harder [3].

Since there is currently no standardized test for recording cognitive fatigue available and biological markers cannot be used to identify these [4], the task was to write an application that should determine cognitive fatigue as part of a study on the basis of daily self-tests.

The application should therefore be able to determine the general condition of a subject with the help of a questionnaire that should be filled out regularly and replace earlier paper questionnaires and be able to save it centrally in a database. In addition, a self-test should be available, which in the best case should be completed several times a day. This should contain two tasks that check the respondent's reaction time and memory ability. In addition, with each run of the self-test, the current emotional state and the current activity of the test person are recorded so that conclusions can be drawn from them later. This information is recorded in a database that is to be exchanged regularly with a server database in order to be able to export all user data centrally for a study.

[1] Shortz, A.E., Pickens, A., Zheng, Q. et al. The effect of cognitive fatigue on prefrontal cortex correlates of neuromuscular fatigue in older women. J NeuroEngineering Rehabil 12, 115 (2015). https://doi-org.ezproxy.hs-

augsburg.de/10.1186/s12984-015-0108-3 [2] Wylie, G.R., Genova, H.M., DeLuca, J. et al. The relationship between outcome prediction and cognitive fatigue: A convergence of paradigms. Cogn Affect Behav Neurosci 17, 838–849 (2017). https://doi-org.ezproxy.hs-augsburg.de/10.3758/s13415-017-0515-y [3] Scott, E., Feeling irritated, stressed, and finding it hard to get stuff done? You might have cognitive fatigue (2019). https://metro.co.uk/2019/02/09/feeling-irritated-stressed-finding-hard-get-stuff-done-might-cognitive-fatigue-8469750/ [4] Price E., Moore G., Galway L., Linden M. (2016) From Paper to Play - Design and Validation of a Smartphone Based Cognitive Fatigue Assessment Application. In: García C., Caballero-Gil P., Burmester M., Quesada-Arencibia A. (eds) Ubiquitous Computing and Ambient Intelligence. UCAmI 2016. Lecture Notes in Computer Science, vol 10069. Springer, Cham. https://doi-org.ezproxy.hs-augsburg.de/10.1007/978-3-319-48746-5_33

Background

Sample headline 1

Sample Text

Sample headline 2

Sample Text

Methodology

This section deals with the development process and how the project topic was approached, as well as the decisions made in relation to the programming languages and environments.

Furthermore, we deal with the architecture of the application, that is divided into stand-alone client area and a server architecture. The data exchange and export plays an important role here, which is also discussed in this chapter and later in the implementation part of the report.

Project management

Overall, the project took place within five weeks and five participants who worked full time on the project. The decision in the area of project management are described in more detail below.

Agile project development with scrum

The project management method SCRUM in a simplified version was chosen for the project. This had the advantage that, on the one hand, programming was agile under certain rules, on the other hand some times and some overhead were reduced.

The Daily Scrum Meeting was an important part of the daily routine, which was occasionally expanded to talk about the current status and follow up on the relevant issues. Due to the limited project time of five weeks, it was decided against the use of sprints and the additional overhead.

A retrospective meeting was also held once a week, which should contain the current problems and suggestions for improvement, but also positive aspects of the project. That helped the project flow and the satisfaction of the participants.

Open Source Software

The soTired project emerged as a university project in which all participants decided to implement an open source solution. This means that both, so the client application code and the server code are publicly available under the license * insert license here *. This means that everyone can view the code, edit it and use it for their own purposes if it can still be made available as an open source solution. This has the advantage that the study participants can see the code and which data is used and stored. This ensures data security. In addition, the code used here is made available to the science and can be further developed. Since the study project is very limited in time, there are some areas of the app that can be added and adapted later on. This can increase the quality of the code in some way.

Architecture

Sample text here

Database management

Sample text here.

User interface

sample text here

Client server communication

sample text here

Implementation

sample text here

Flutter application

Sample text here.

Go backend

sample text here.

Results and discussion

Sample headline 1

Sample Text

Sample headline 2

Sample Text

Conclusion

Sample headline 1

Sample Text

Sample headline 2

Sample Text

References

[1] Test