Designing a Survey Tool for Monitoring Enterprise QoE

Kathrin Borchert
Chair of Communication Networks
University of Würzburg
Germany
kathrin.borchert@informatik.uni-wuerzburg.de

Matthias Hirth
Chair of Communication Networks,
University of Würzburg
Germany
matthias.hirth@informatik.uni-wuerzburg.de

Thomas Zinner
Chair of Communication Networks,
University of Würzburg
Germany
zinner@informatik.uni-wuerzburg.de

Anja Göritz
Occupational and Consumer Psychology,
Freiburg University
Germany
goeritz@psychologie.uni-freiburg.de

ABSTRACT

Enterprise applications like SAP are part of the day-to-day work of a large number of employees. Similar to many modern applications, enterprise applications are often implemented in a distributed fashion and consequently suffer from network degradations resulting in impairments like increased loading delays. While the influence of these impairments on the perceived quality of users is well researched for consumer applications and network services, the impact of these impairments in a business environment is yet to be investigated. To address this gap we develop a non-intrusive software tool for continuously collecting subjective ratings on the performance of an enterprise application from a large number of employees. Based on the feedback from a company and results from two initial field studies we discuss the specific challenges when assessing the perceived quality of employees during regular working hours and point out our further research directions.

CCS CONCEPTS

• **Networks** → *Network monitoring*; • **Human-centered computing** → *User studies*;

KEYWORDS

Quality of Experience, Enterprise, User Study

ACM Reference format:
Kathrin Borchert, Matthias Hirth, Thomas Zinner, and Anja Göritz. 2017. Designing a Survey Tool for Monitoring Enterprise QoE. In *Proceedings of Internet QoE '17, Los Angeles, CA, USA, August 21, 2017,* 6 pages.
DOI: http://dx.doi.org/10.1145/3098603.3098610

1 INTRODUCTION

Employees in several business areas have to work with applications and services to fulfil their day-to-day work. These applications run on top of distributed systems, like thin client architectures. Thus, the application performance may suffer from congestion and performance issues originating in the network as well as in the data center. This leads to noticeable delays and other impairments at the user side, hence decreasing application performance, which may influence the perceived application quality of the employees.

To build an objective model based on the technical parameters of an application, e.g. the processing time at the server or the overall response time, the impact of these parameters on the quality of experience (QoE) of the users has to be investigated. An approach for identifying influence factors in the enterprise context is to measure the QoE of the users and simultaneously analyze the technical performance data of an application. However, several new challenges arise when performing QoE measurements in enterprise environments. Beside cost factors, the integration of such measurements in the enterprise infrastructure and in the day-to-day work of employees need to be taken into account. As the QoE study is running in the live system to gather the rating of the application performance in a realistic environment, the measurements need to be non-intrusive and must not interrupt critical working processes of the employees.

This paper tackles these challenges by designing and implementing a survey tool for enterprise environments and demonstrating its feasibility. For this, key requirements of the tool for assessing the perceived quality of enterprise applications are deduced in cooperation with a large company. Based on these requirements, we developed a survey tool and integrated it in the IT infrastructure of the cooperating company. In two large pilot studies lasting several weeks the applicability of the approach is demonstrated.

The remainder of this work is structured as followed. Section 2 outlines the related work on the impact of system impairments on the users' QoE and approaches for measuring enterprise QoE. Section 3 discusses the requirements for conducting such measurements in enterprise environments and the realization is described in Section 4. Section 5 presents the results and the feedback of the user studies, Section 6 concludes with an outlook of possible directions of future work.

2 RELATED WORK

Numerous studies exist which focus on the identification and quantization of factors influencing a customer's or end-user's satisfaction

Internet QoE '17, August 21, 2017, Los Angeles, CA, USA

ISBN 978-1-4503-5056-3/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3098603.3098610

with the performance of applications and services. One of the main influence factors, and thus an extensively studied aspect, in network based applications and services, is the transmission delay and the resulting application behavior. Several studies investigate, e.g., the influence of loading delays [1, 6, 13, 14] on the QoE of web site users. The results show the relation between network and application parameters and the users' QoE and derive thresholds for the usage of these services. It can also be shown that the impairments directly affect the user behavior [4, 11, 13] and result in a, e.g., lower number of system interactions.

Until now, most QoE studies focus mainly on (web-based) applications and services for end-users but enterprise applications are mostly neglected. Even if findings from existing studies give an impression how impairments like delays influence the perceived quality of users of interactive systems, it is not clear how the system performance and especially delays are perceived by employees using network-based applications in their day-to-day work. However, especially in this context delayed systems and un-satisfied users, i.e. employees, might lead to potential decreases in the productivity, which might directly result in financial drawbacks.

A first indication of the influence of delays in the context of business software is given by Bonhag et al. [2]. They investigated the perceived quality of a fictive business application by emulating loading delays. The results show that the QoE is affected by a delayed application performance. The study is conducted online, thus the environment is uncontrolled and additional factors, like network delays, may influence the results. Although, the participants are working everyday with business software, the experiment does not consider the context of the usage of the software.

One approach to assess the perceived system quality in an enterprise is using the feedback and information given by employees via existing communication channels in an enterprise, e.g. by analysing the messages of a ticketing system [16] or by considering system reports and requests for assistance after a software release [9]. However, these methods only provide very coarse-grained data and the evaluation is often difficult due to the unstructured information in the support requests.

Approaches resulting in finer grained data usually involve active user feedback during or immediately after the use of the service or application. Examples for studies using this approach are Schlosser et al. [10] and Casas et al. [5]. Both works aim at a better understanding of the influence of varying technical parameters on the QoE of enterprise and related tasks like typing on a thin client. However, the tests were conducted in dedicated labs with students and not in a working environment with employees. Whereas, Smith et al. [12] introduced an approach to collect feedback directly from employees in a real business environment. The feedback about the performance of the meeting software Microsoft Lync is collected at the end of each session via a survey realized as a game. While this approach is feasible for a software that is only used from time to time, it cannot be applied for applications that are used throughout the whole workday, like e.g. SAP systems, as it would be too time and cost intensive to ask the users to rate the performance after each interaction. Further, it is also not practicable to reduce the costs by collecting ratings only once a day, e.g. after the last system interaction of a working day, since the last interaction is not known in advance.

3 SURVEY TOOL REQUIREMENTS

In the following, we discuss the requirements for a tool used to assess the perceived application quality of employees continuously and at a large scale. The requirements are derived from discussions with a cooperating company and feedback of the participants of two pilot studies.

3.1 Minimization of costs

The main difference between lab-based studies for assessing perceived application quality and assessments in an enterprise production environment is the context of the participants. While participants in a lab study solely focus on the assessment tasks, employees need to focus on their regular day-to-day work and the assessments impose additional work. Consequently, one requirement is the minimization of the effort for each participating employee. A possible solution is the limitation to a small number of questions and required interactions per assessment. This can be realized by focusing on an interface with selectable items, e.g., checkboxes or radio buttons, instead of free text answers.

In addition to the assessment time, the number of participating employees needs to be minimized. Even if an employee can complete one assessment within a few seconds scaling out the assessment process to all company employees can result in a significant amount of working hours per year. Thus, the number of participants needs to be dimensioned appropriately to generate representative results but also limit required man power.

3.2 Communication concept & training phase

Similar to other studies, the participants need some basic instructions regarding the survey process. Due to the large number of participants, a personal training is not possible and similar challenges to training phases in crowdsourced subjective studies arise. Further, the participation in the study is not mandatory, thus an appropriate motivation needs to be provided during this initial communication with the employee. Additionally, also the duration of the training phase needs to be kept as short as possible.

3.3 Seamless integration

During the assessment of the perceived quality, the employees still have to complete their day-to-day work. This might include cognitive challenging tasks or personal contact to customers of the enterprise. Consequently, the survey tool needs to be seamlessly integrated into the existing workflows, or at least the imposed disturbance needs to be minimized. Further, unlike most other subjective test scenarios, the survey tool does not run on a dedicated test or evaluation system but needs to be integrated into the production system of the company. This imposes the need for additional security considerations, error handling and also limits the available technologies. The ratings of the users submitted through the tool have to be considered as privacy relevant data and have to be stored in a secure manner. Appropriate means have to be taken to anonymize the identity of the employees or strict regulations for the accessing the data nave to be added. Software errors potentially

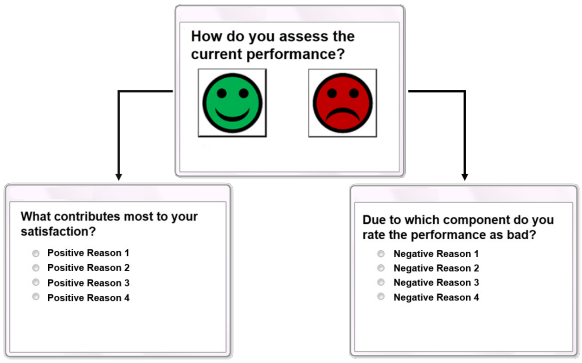

Figure 1: Graphical user interface.

affect a large number of employees, thus the main requirement for the error handling is that the software should fail gracefully. This means that in case of a software error, the software should be terminated without any notice of the user, even if this implies a loss of measurement data. Finally, most companies maintain a given software and infrastructure stack that defines the available technologies for the development of the survey tool.

3.4 Common best practices for user studies

Despite the previous requirements and design limitations, the assessment methodology needs to be scientifically valid and respect established standards for subjective evaluations like ITU-T P.913 [8]. Additionally, it should follow common best practices, e.g., no implicit or explicit incentives for the employee to give a certain rating.

4 SURVEY TOOL REALISATION

Based on the defined requirements, we designed and implemented a simple tool for monitoring the QoE of employees using active feedback. In the following, we discuss the concrete implications for the design and implementation of the tool, as well as the communication concept for the participating employees.

4.1 Interface design

The QoE monitoring is realized as a survey which is shown via a pop-up to the participating employees. The survey comprises two steps which are shown in Figure 1. First, the user rates the system performance by clicking on a happy green colored or an unhappy red colored smiley. The green smiley represents an acceptable or good performance and the red smiley defines an unacceptable or bad performance. We intentionally omitted a "neutral" option to correlate the ratings with technical measures and determine an acceptability threshold for those measures in future work. After rating the performance, the user may explain his rating by selecting one out of several predefined reasons. The application supports the functionality to customize these reasons and create different subset for specific groups of employees. To fit the needs of all participating groups, the possible reasons should be defined in cooperation with experts from the enterprise that already identified potential influence factors on the application's performance. Nevertheless, even if the set of reasons is optimized for the requirements of the specific group, not all possible performance issues can be taken into account. Thus, it is advisable to also add a reason "other", in case none of the predefined reasons fits for the user.

4.2 Minimization of assessment time

To reduce the completion time of the survey, the number of questions is limited to a maximum of two. Further, the answer of each question requires only one click. After rating the performance by clicking on one of the smilies the second question appears automatically. The interface of the survey tool is also automatically closed as soon as a reason is selected in the second step. Hence, the completion of the survey requires two clicks. We do not allow multiple choice in the second step although this has been requested in the participants' feedback of the pilot study, as this requires an active submission of the survey resulting in a minimum effort of three clicks. For similar reasons we neglected the integration of an input field to enter an individual reason for the selected performance rating or other additional information. Furthermore, text fields entail the risk that users enter sensitive customer data accidentally as observed in the pilot study with the first prototype of the tool.

Whereas the omission of selecting a reason for a positive rating would further reduce the number of required clicks and the completion time of the survey, this might also encourage the employees, to select the "faster" path through the survey. Thus, the second survey step is mandatory in both, for positive and negative ratings.

Besides the number of required clicks, we optimize the assessment time by using colored icons instead of text buttons for the rating step. This reduces the amount of text in the pop-up and the colored smilies are easier to identify.

4.3 Integration into day-to-day work

As discussed previously, it is not possible to ask the employees to rate the system performance after each interaction with the system. Instead, the pop-up opens automatically once an hour if the user is logged into the system and the user rates the system performance within the last hour. Binding the pop-up timer on additional thresholds, e.g., a minimum number of interactions of the user with the system, would require a tight connection of the survey tool with the production system of an enterprise and thus is not desired in this context. To prevent the interruption of critical working processes or conversations with customers the pop-up is also closed automatically after a few seconds if the user does not react. These ratings are marked as missing. Thus, it is possible to distinguish between ratings which are marked as missing and those which are not given because the user is not logged-in the system. If the user only rates the system performance and does not select a reason, the pop-up is also closed after a specific amount of time. In this case, the rating is stored and the reason is marked as missing.

4.4 Implementation

The tool comprises a client and a server component. The client side is written in C# and is automatically launched after logging into the system. As mentioned before, it is very important that software or configuration errors do not affect the employee in the daily work. Thus, the client component only supports a text-based error log but does not display any notification to the employee.

Furthermore, before opening the pop-up, the client component sends a verification request to the server component including a predefined ID for the employee. The pop-up is then opened only upon confirmation of the server. This allows a remote administrator to easily stop the survey as a whole or for individual employees. After displaying the pop-up on the client side, the server calculates the next time the pop-up should be opened. The client software is put to sleep in order to save resources and again sends a pop-up verification request at the given point in time. In a preliminary version of the survey tool, the time difference between two pop-ups was set exactly to one hour. This resulted in the effect that the participants were "expecting" the pop-up and prepared themselves to rate the performance. Thus, the ratings were not spontaneous and in some cases working groups coordinated their responses. We prevent this side effect by varying the interval between two pop-ups while considering a minimal interval of 15 minutes and a maximum interval of 119 minutes.

The server component is realized with a PHP framework. Beside the communication with the client component it provides several functionalities for the configuration and the management of the surveys, e.g., the management of the predefined reasons and the groups of participants. The back end also provides the possibility to add participants manually or to generate a representative sample of employees by a simple random sampling mechanism.

Another purpose of the server component is the communication with the database to store the responses of the participants. Beside the ratings and the reasons additional information as the time when the survey was opened at the client and the time stamp of submitting the response is stored. The time stamps are required to determine the next rating time as well as to analyze the response behavior, e.g. concerning the time needed to complete a survey.

4.5 Study execution

User studies in general include a training-phase, e.g., a brief introduction for the participants. The same is required in this context, however the remote location of the participants and the business setting have to be considered. Thus, we applied the following communication concept to inform the participants and to coordinate the conduction. A few days before starting the survey the participants are informed via email including information like the respective starting date of the survey and its duration. The email also includes background information concerning the survey and the instructions, e.g. how to use the tool and that the users should rate the system performance within the last hour on a subjective and individual base, instead of coordinating their feedback. Beside a brief explanation of the overall goal of the monitoring, background information including more details about the design of the software, e.g. why the users have to select a reason for positive ratings, is given. The detailed information was included upon request of the initial users during the pilot study and to motivate the users, as they realize active participation can be used to improve their working conditions. Beginning with the announced starting date, the survey is conducted which means the pop-up is shown once an hour to the participant during the previously specified period. At the end of the survey period another email is sent to the participants, thanking for their participation and asking for further feedback regarding the conduction as well as possible ways to optimize the tool.

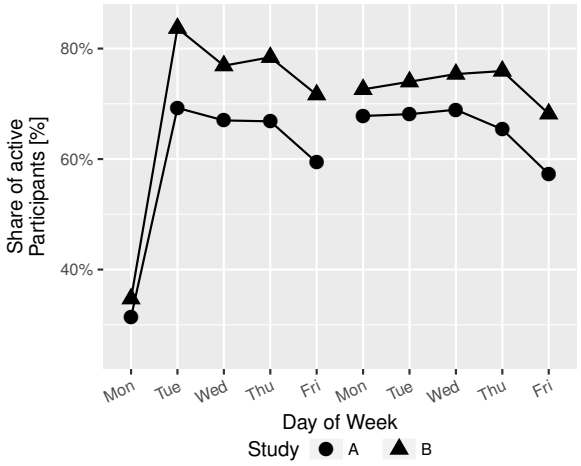

Figure 2: Share of participants submitting at least one survey per weekday.

5 USER STUDIES

We evaluate the applicability of our tool by two user studies in cooperation with a company with more than 15000 employees. The feedback has been collected during two working weeks in study A from 618 participants in December 2015 and in study B from 723 employees in January 2016. The list of possible reasons for negative ratings was created in cooperation with experts of the enterprise and included performance issues in a set of software modules required for the employees' day-to-day work and the option "other". Due to the different tasks of the participants in both studies, the list of modules in study A and study B differs.

5.1 Integration into day-to-day work

In total the survey has been shown in study A 33225 times with 16339 ratings marked as missing and in study B 47113 answers have been collected with 23525 marked as missing. This indicates that it is not always possible for the employees to answer the survey during the daily working process. Reasons for this may be the completion of time critical work, talking with customers, or absence from their working place. Despite the percentage of missing answers the results show about 97% of the participants submit at least one survey during the total survey period of two working weeks. Figure 2 highlights the share of active participants per weekday compared with respect to the total number of participants included in the study. We define a participant to be active, if he submits at least one rating within a day. The different markers represent the two studies. The share of active participants ranges from 31% to 69% for study A and from 35% to 85% for study B. Possible reasons for the higher number of participants for study B are the lower number of part-time employees in this user group as well as the fact that these employees mainly focus on data processing and are less involved in customer care. Due to a ramp-up phase at the start of the study, the number of active participants on the first day is

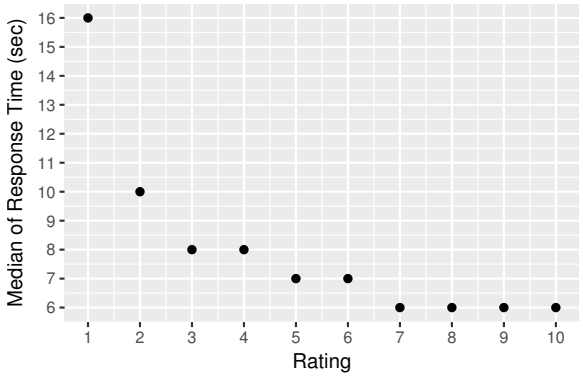

Figure 3: Time needed to rate the performance.

lower than on the other days. Indicated by the highest measured share of participants during the two weeks, the ramp-up phase is finished on the second day. Comparing the remaining days, there are lower values measured at the end of each week. On Friday, the office hours in the enterprise are usually shorter than during other working days. This, in conjunction with a number of participants working only part-time, leads to less employees participating in the survey. Except for the start phase of the study the two weeks do not differ significantly overall. Due to the continuous active feedback collection once an hour the response rate per day is significantly higher than in other approaches with only a single response. Here, response rates less than 30% are often observed for surveys conducted in enterprises [12].

5.2 Assessment time

In order to evaluate the additional effort imposed to employees we investigate the overall response duration for study A. Missing ratings and ratings without reasons are omitted. In total, 95 % of the employees submitted at least one rating and the median response time of the participants is 6 seconds. Due to outliers with a response time of 132 seconds, the mean response time is significantly higher with 9.6 seconds. To analyze the speed-up in answering the pop-ups over time, we focus on employees that submitted at least 10 ratings. This applies to 79 % of all participants and their median answering time for the first 10 ratings is shown in Figure 3. The rating times are measured at an accuracy of one second, as this is sufficient for an estimation of the additional effort imposed by the survey tool. The figure indicates that during the first 7 ratings, the response times are decreasing from 16 sec to 6 sec, while they remain roughly similar at 6 sec after the 7th rating. Due to non-normally distributed response times and repeated ratings for the same employees we use the Friedman's test [7] to confirm the changing user behavior. The test clearly shows that response times for the first 7 ratings significantly differ ($\chi^2(6) = 426.81, p < 0.001$), and that response times for the 8th to 10th rating are not significantly different ($\chi^2(2) = 0.15, p > 0.05$). Overall, we can assume that at the beginning of the survey new participants need to get used to the survey questions and the interface. In the later course of the survey, the participants can answer the questions more easily and efficiently.

We use a Wilcoxon signed-rank test [15], to check if response times differ between positive and negative ratings. The input data is aligned as follows. Each employee e submitted m positive ratings $r^+_{e,1}, \ldots, r^+_{e,m}$ and n negative ratings $r^-_{e,1}, \ldots, r^-_{e,n}$. For the evaluation we thus consider for each employee e the ratings R_e with $R_e = \{r^+_{e,1}, \ldots, r^+_{e,k}, r^-_{e,1}, \ldots, r^-_{e,k} | k = min(m,n)\}$ and the corresponding rating times. This results in total in $\sum_{e \in E} |R_e| = 6204$ considered rating times, with E being the set of all employees who submitted at least one rating during the study. Based on this subset, the difference between rating times for positive ratings and the rating times for negative ratings is not significant with $p > 0.05$.

The importance of a time efficient assessment process can easily be demonstrated by considering the total time t_t spend on the active submissions. The total time t_t can easily be calculated as the sum of all response times which results in $t_t \approx 38$ hours. This also shows an estimation t'_t of the total time t_t based on the median response time of 6 sec and the total number of active submissions $s_a = 47113 - 23525 = 23588$ in study A seams feasible, as $t'_t = 6\,\text{sec} \cdot s_a \approx 39$ hours.

5.3 User Feedback

At the end of each survey we collected optional feedback from the participants concerning the survey in general, the interface of the application and the content of the survey. In the following we discuss the merged feedback from study A and study B.

5.3.1 *General feedback.*

- The employees are satisfied that the company is interested in their opinion.
- The survey receives in general neutral or positive acceptance.
- "Every time the pop-up has been shown everything works fine."

The general feedback confirms the acceptance of the application by the users as discussed in Section 5.1. It shows that the users are willing to answer the survey and that they do not feel disturbed by the pop-up. Further, we find psychological side effects. The participants suggest that there are no impairments if the tool is running. Nevertheless, it is not possible that the application influences the performance of the system.

5.3.2 *Feedback on conduction.*

- The survey duration of two weeks is appropriate.
- The provided information for the participants is sufficient and understandable.

The feedback concerning the duration shows that the employees accept time periods of two working weeks. We suggest not to ask the employees during a shorter time period because here it is difficult to observe additional influences on the performance, e.g. effects from software updates, peak and off-peak times of the employees.

5.3.3 *Feedback on interface and content.*

- The tool is easy to use.
- Sometimes it is not clear if the participants should rate the system performance between two pop-ups or the performance of their last interaction with the system.
- Some of the predefined reasons are ambiguous.

- The predefined reasons do not match the requirements of all working groups.
- The participants would like to provide additional information when choosing the reason "other".

The stated irritation about the rating is not caused by the design of the interface of the application as the users find it easy to use. Instead, it confirms the importance of providing sufficient instructions to the participants. Here, we have to highlight the instruction to rate the system performance intuitively.

The analysis of selected reasons for negative ratings shows a large amount of ratings explained with the reason "other". It is about 58.5% for study A and 44.7% for study B. This result confirms the feedback of the participants that the predefined reasons do not match the requirements of all participants. On one hand, it shows that it is difficult to fit the needs of all employees from different working groups or branch offices with a limited number of reasons. On the other hand, it indicates that there may be additional factors which influences the perceived performance quality of the system, e.g. other system components that were not considered to be performance critical or usability aspects of the software. A possible solution to gather those missing reasons directly from the participants via an additional communication channel, e.g. via e-mail or a discussion forum, during or at the end of the survey period. Due to the flexible implementation of the survey software, those reasons can easily be added to follow-up studies.

6 CONCLUSION

In this work we discussed the challenges and specific requirements for measuring the perceived application quality of employees in an enterprise environment. We introduced a survey tool for rating the performance of enterprise applications which considers these requirements, e.g. the minimization of the assessment time. The practicability of the tool is evaluated by two large user studies with hundreds of participants.

The studies show that the survey is seamlessly integrated into the day-to-day work of the participants which is reflected in the amount of ratings and the short median assessment time of 6 seconds. Further, the collected feedback of the participants and the evaluation of the results of the user studies show several interesting side effects. One discovered side effect is that several participants stated that they perceive a better application performance when the survey window pops up. Hence, it might be beneficial to extend the tool to allow push-based user ratings at arbitrary times. These responses should be separated from the regular ratings and might be used to identify performance problems on short time scales. Further, the feedback also indicates the complexity of finding an appropriate subset of positive/negative reasons to better understand the context of the participants. This might be improved by an additional survey allowing to gather relevant technical and non-technical reasons for the specific user group. Nevertheless, the aggregation of possibly unstructured input and the selection of appropriate reasons still requires domain-specific expert knowledge.

However, to make full use of the collected data, a holistic workflow needs to be designed that enables enterprises to automatically identify the reason for a decreasing employee satisfaction and take appropriate countermeasures. This is still part of ongoing research.

Another line of research is the interconnection of the collected user feedback with monitoring data of technical parameter to build QoE models for specific enterprise applications. For that, several constraints of the survey tool have to be taken into account. The varying time interval between two ratings may result in a varying amount of interactions with the applications and thus with the technical system. Further, interactions completed right before a rating may influence the rating to a higher extent than previous interactions. This effect can be mitigated by either reducing the time between ratings or by enabling push-based user ratings as outlined above. Other solutions are to aggregate response times on a regular basis or to weight the impact of the transactions on the user ratings depending on the distance. A first step towards combining user feedback and technical monitoring data in enterprise environments is discussed in [3].

ACKNOWLEDGEMENT

The authors express their gratitude to André Rentsch, Peter Karg and Harald Wehnes for the support during the course of this work. This work is supported by the Deutsche Forschungsgemeinschaft (DFG) under Grants TR 257/38-2 and by kubus IT. The authors alone are responsible for the content.

REFERENCES

[1] Ioannis Arapakis, Xiao Bai, and B. Barla Cambazoglu. 2014. Impact of Response Latency on User Behavior in Web Search. In *ACM SIGIR Conference on Research & Development in Information Retrieval*.
[2] Wolfgang Bonhag, Doreen Feindt, Siegfried Olschner, and Ulf Schubert. 2015. Wie schnell ist schnell bei Business-Software? Analyse zur Performance bei der Nutzung von Business-Software. *Mensch und Computer–Usability Professionals*.
[3] Kathrin Borchert, Matthias Hirth, Thomas Zinner, and Decebal Constantin Mocanu. 2016. Correlating QoE and Technical Parameters of an SAP System in an Enterprise Environment. In *International Teletraffic Congress*.
[4] Jake Brutlag. 2009. Speed Matters for Google Web Search.
[5] Pedro Casas, Michael Seufert, Sebastian Egger, and Raimund Schatz. 2013. Quality of experience in remote virtual desktop services. In *International Symposium on Integrated Network Management*.
[6] Sebastian Egger, Tobias Hoßfeld, Raimund Schatz, and Markus Fiedler. 2012. Tutorial: Waiting Times in Quality of Experience for Web based Services.
[7] Milton Friedman. 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association* 32, 200 675–701.
[8] ITU-T. 2016. Recommendation P 913. *Methods for the subjective assessment of video quality, audio quality and audiovisual quality of Internet video and distribution quality television in any environment*.
[9] Audris Mockus, Ping Zhang, and Paul Luo Li. 2005. Predictors of customer perceived software quality. In *International Conference on Software Engineering*.
[10] Daniel Schlosser, Barbara Staehle, Andreas Binzenhofer, and Björn Boder. 2010. Improving the QoE of citrix thin client users. In *International Conference on Communications*.
[11] Eric Schurman and Jake Brutlag. 2009. Performance related changes and their user impact. In *Velocity – Web Performance and Operations Conference*.
[12] Ross Smith and Lori Ada Kilty. 2014. Crowdsourcing and Gamification of Enterprise Meeting Software Quality. In *International Conference on Utility and Cloud Computing*.
[13] Stoyan Stefanov. 2008. YSlow 2.0. In *China Software Developers Network*.
[14] Dominik Strohmeier, M. Mikkola, and Alexander Raake. 2013. The importance of task completion times for modeling web-QoE of consecutive web page requests.
[15] Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin* 1, 6 80–83.
[16] Thomas Zinner, Florian Lemmerich, Susanna Schwarzmann, Matthias Hirth, Peter Karg, and Andreas Hotho. 2015. Text Categorization for Deriving the Application Quality in Enterprises Using Ticketing Systems. In *International Conference on Big Data Analytics and Knowledge Discovery*.