

The system for control and accounting of working time

A. Plakhtii, V. Shkarupylo, S. Byelikov

Abstract—The comparative analysis of existing timesheet systems for creating a number of requirements to own solution has been conducted. Software requirements specification for working time control and accounting system (time tracking) has been created. The dependencies between the components of proposed solution have been analyzed. For this purpose, the Electron framework has been used. Basing on the results of the analysis conducted the system for working time control and accounting has been proposed. The proposed system is a cross-platform solution that allows to control and account the activity of employee during the working time.

Keywords—Timetracker, Timesheet Software, Working Time, Cross-platform.

I. INTRODUCTION

Working time is one of the most significant conditions of employment contract. Daily accounting positively affects the projects control aspects: it's much easier for project leaders to control the tasks accomplishments and the appropriate costs, to adjust the budget just in time (on the fly), with respect to current needs – not on the final stage of project-related work, when it's obviously that it's impossible to finish in time. Even the companies taking the fixed fee for the project work are beneficial here – they can easily analyze which categories of work and which clients are more preferable for them. The considered theme is especially topical for the people willing to work remotely and with respect to their own schedule – the freelancers [1].

The main task of working time control and accounting systems is to increase the business processes effectiveness, aimed at employees' work effectiveness increase – to give them the ability to establish the flexible schedules for themselves, with no time losses [2].

The working time accounting can be built on a daily, weekly or summarized basis. Nevertheless, in each of these cases the time spent on work per day is considered [3].

To estimate the peculiarities of employees' work effectiveness, to get the complete picture of how the working time is actually being spent, the software of the following classes is widely used: the time tracking software, the task management software and the PPS-systems (Personal Productivity Systems) which are also known as GTD-systems (Getting Things Done) [4].

There are the following reasons for time tracking systems usage: to involve the additional resources; to increase the accuracy of time utilization reports, to force the employees to be more responsible; to distinguish productive employees from the rest; to foster the new projects planning; to provide the control of remote freelance employees [5].

II. PROBLEM STATEMENT

The problem of employees' working time accounting becomes more and more topical each year, because to ground the illicit activities, carried out by the staffers during the working time, the continuous control has to be brought to the table [6]. Moreover, such control is intended to stipulate the employees to do their work more thoroughly and in more coordinated

A. Plakhtii Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: n.plakhtii@gmail.com).

V. Shkarupylo Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: shkarupylo.vadym@gmail.com).

S. Byelikov Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: rector@zntu.edu.ua).

manner. Because of the fact that workers can operate with different operating systems, the problem of such universal cross-platform control and accounting system development arises.

The key aspect here is to diminish the costs intended to be associated with system tweaking with respect to the specific requirements of certain operating system. This peculiarity is a distinguishing feature of our work. To this end, the following tasks are resolved in our work: to conduct the comparative analysis of existing time tracking systems; formulate the requirements to own solution, with respect to the results of comparative analysis; to answer the question if our solution is actually a valid one – for this purpose the validation has to be conducted – by way of testing.

III. COMPARISON OF SIMILAR EXISTING SOFTWARE PRODUCTS

There are plenty of different time tracking systems. The typical one is a Timesheet Software – actually a table which should be manually filled on a daily basis – with a list of accomplished tasks, but only in electronic format [7]. Such software is typically installed locally, with a limited set of reports. It is applicable for small and medium companies in case the specificity of their activities doesn't cover numerous distractions, bounded with Internet and social networks in particular [8].

In a large companies there is a totally different picture – the vital need for Time Tracking (Recording) Software takes place. The appropriate systems should also be characterized with the following features: to provide the abilities to gather the information about the operations being performed by the employees at their workplaces; be integrated with modern project management systems [9]. An attempt to compare the feature sets of existing time tracking systems is given in Table 1.

TABLE I
THE COMPARISON OF MODERN TIME TRACKING SYSTEMS

Title Feature set	Harvest	Toggl	Rescue Time	Log My Hours	Prima ERP	Upwork
Manual / Automatic modes support	+	+	-	+	+	+
Differentiation between commercial and non-commercial projects	+	+	-	+	-	-
Automated reports generation	+	+	+	+	+	+
Creation of flexible invoices	+	-	-	+	+	+
An autonomous mode of timers functioning (automatic synchronization)	-	+	+	-	+	+
Functioning in a background with an active monitoring of application	-	-	+	-	-	-

Taking into consideration the content of Table 1, we assume that the most important functional properties of our system should be the following ones: the support of both – manual and automatic – modes of working time accounting; automatic reports generation with respect to defined time intervals; new projects have to be classified as commercial (profitable) and/or non-commercial (non-profitable) ones; in case of failures, the application has to be synchronized automatically.

Time tracking systems are accessible in various forms: local clients for Windows, Mac and Linux platforms; mobile clients; web services; extensions for web browsers; accounting systems for small business; free products and licensed software.

IV. SOFTWARE REQUIREMENTS SPECIFICATION

In accordance with software systems requirements specification drafting technique, proposed by IEEE (IEEE 830-1998 Standard), appropriately created specification has to provide customer with rigid and clear description of what is required to be done. For the executors such specification has to give the clear understanding of what the customer is expecting. For this purpose, the SRS (Software Requirements Specification) draft has to be elaborated. The structure and the content of concrete requirements specifications have to be defined. The additional supplementary documents should also be created, e.g. control letter for SRS quality checking and/or regulator's handbook [10].

The requirements specification for the system is briefly described below.

Introduction. The requirements specification describes functional and non-functional requirements to time tracking system. This document is devoted to be used by the company, responsible for system functioning correctness checking [11].

Our system is named as «TickTock». This system should allow the companies with about 100 employees, not taking into consideration the specificity of work, to account and control the activities of their staffers during the working time.

Time tracking system has to support the following functionality: the invoices accounting; to create the reports on weekly basis; to maintain the convenience of project management by way of task selection. To maintain these functionalities, the SMTP protocol (Simple Mail Transfer Protocol) has to be implemented. This protocol allows to transfer e-mails to postal server or from client computer, or between postal servers [12].

The further information is about the specifics – specific functional requirements and system limitations.

Detailed description. Our «TickTock» system is a plain time tracking system intended to be used by companies with about 100 employees, and their work has to be carried out on personal computers.

The system being developed reads and stores the metrics from workers' personal computers and builds the statistics from their activities, and also synthesizes the reports in form of electronic letter, then these reports are ought to be sent to the recipients with respect to the content of specified e-mails list.

User interface is represented with computer mouse controller, graphical interface with additional information and control elements: the information about working hours (spent / to be spent); tracking switch-on and switch-off buttons.

Software interface is represented with SMTP Gmail server – to send the messages (reports) to specified recipients.

Communication interfaces – with support of HTTP (Hyper Text Transfer Protocol) and SMTP protocols – to use the broadcast services.

Memory volume requirements – not less than 200 MB of free space on hard disk drive.

The usage requirements – the system has to be adopted for popular operating systems.

System functions. The system switches-on/off by a double clicking on utility icon. The first stage to get started is to log-in. After checking the login and password the following system's features will be available for the user: the time tracking start/stop button; the timer showing the time elapsed in hh:mm format; the ability to browse personal working statistics. If certain user is a super-user (with additional privileges), the following abilities are also allowed: to edit the information about users; to browse the statistics about users; to create the list of electronic addresses to send the statistical information about the employees' work; to edit the system settings (time zone settings, the limits of time intervals within which the tracking has to be conducted).

User characteristics. The client is a user (worker) which should have basic skills to work with basic windows applications.

The limitations. System can be utilized only by a registered user. The users can be registered only by a super-users – the administrators.

To make company be able to increase the number of registered employees the login and password have to be previously created for the first super-user. The system can be used on Windows and Linux operating systems. The Internet connection is a mandatory requirement.

Specific requirements. Only the registered user is allowed to be logged in.

Functional requirements to super-users: the ability to store time stamps when system is being switched on/off (after a specified period of time the snapshot of user screen is being taken and stored in a database); the availability of CRUD-functions set (create, read, update, delete) – to browse users' statistics, to shift time intervals; to modify the list of electronic addresses.

Functional requirements to users: to save the time when system is being switched on/off; when specified time is elapsed, the screen snapshot has to be taken and stored in database [13].

Quality requirements. From the safety viewpoint – the users authorization and authentication are mandatory [14].

V. SYSTEM IMPLEMENTATION AND TESTING

One of the most popular solutions for cross-platform applications development is Node.js framework, paired with Electron framework [15]. These frameworks can be easily paired up with any front-end JavaScript solution [16]. Our time tracking system has been implemented on this basis (fig. 1).

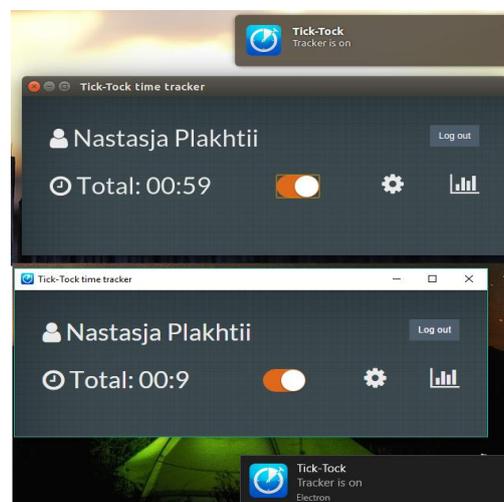


Fig. 1 Snapshot of the interface

In fig. 1 the snapshot of a system running on Ubuntu LTS 14.10 and Windows 10 PRO platforms is given to demonstrate the universality of our solution (to fulfill the universality requirement).

The intense time tracking system utilization in corporate environment is tightly bounded with appealing fault tolerance requirements [17]. The simultaneous connections to system by number of users can entail the faults in system functioning. To this end – to check the liveness properties of system – the Smart Bear LoadUI tool has been used [18]. This tool has been utilized to conduct the validation, and, thus, to prove the validity of system. The information about simultaneous connections and corresponding metrics are given in fig. 2.

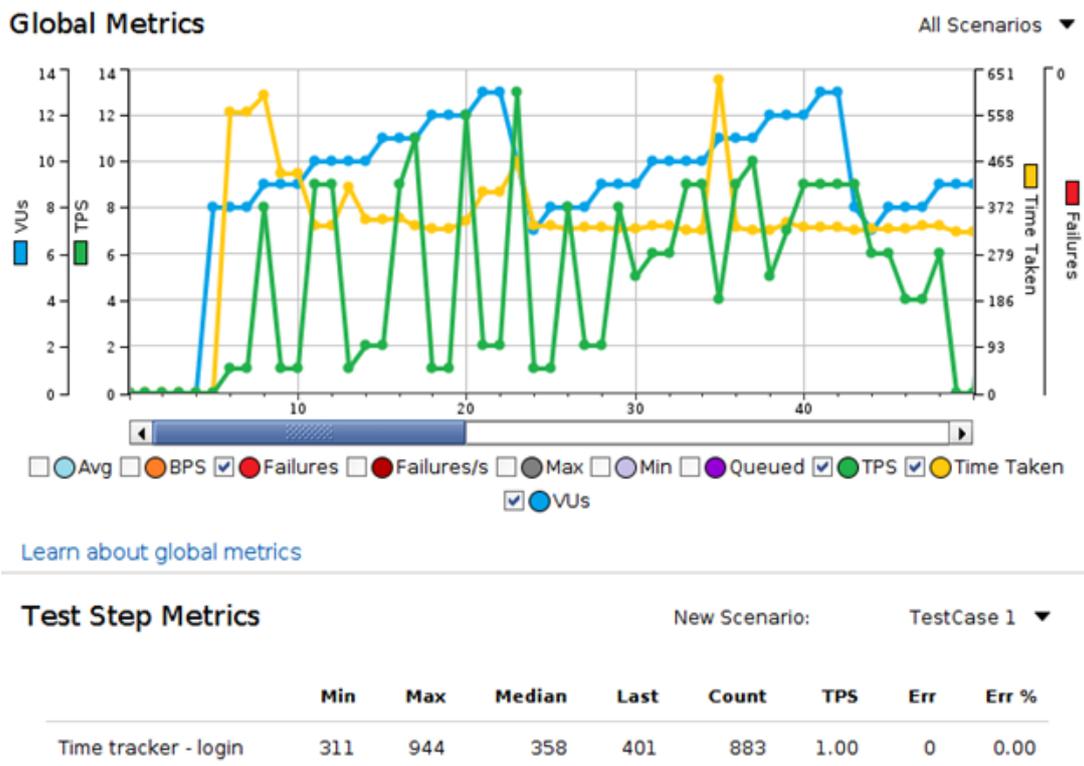


Fig. 2 Global metrics of testing process

In fig. 2 the following metrics can be seen: the maximum TPS (transactions per second) value – 13; Time Taken – the time (in milliseconds) taken for the requests (often comparable to response time values), the maximum value is about 620, the averages are about 300 ms; Failures – the number of failed requests – 0; VUs – the number of simultaneously running virtual users – 13 users. Taking into consideration the obtained results, it can be concluded that fault tolerance requirement is fulfilled and thus the proposed time tracking system can be characterized as valid one.

VI. CONCLUSION

Thus, the following tasks have been resolved:

1. The comparative analysis of existing control and accounting systems has been conducted. Basing on its results, a conclusion about system's most important functional properties has been formulated. The following properties have been distinguished as more important ones: the support of both – manual and automatic – modes of working time accounting; the ability to automatically generate reports for a given time intervals; the ability to differentiate between commercial and non-commercial projects; the automatic synchronization of application in case of failures.

2. Software Requirements Specification has been proposed. This allowed to define the requirements and limitations to be taken into consideration during the development process, e.g., universality, liveness.

3. The cross-platform solution has been developed. For this purpose the Electron framework has been used. The universality requirement fulfillment has been proved under Windows and Linux platforms.

4. The liveness requirement fulfillment has been proved by way of testing. It has been shown that proposed system is fault-tolerant and can be applicable to the scenarios with an intense workload – when multiple users (about 100) interact with system simultaneously.

Further work is aimed at integration of proposed solution with existing project management systems.

ACKNOWLEDGMENT

This publication is the result of the project implementation: TEMPUS CERES: Centers of Excellence for young REsearchers (Reg. no.544137-TEMPUS-1-2013-1-SK-TEMPUS-JPHES).

REFERENCES

- [1] Bundymuseum.org, “Willard Legrand Bundy Biography”, 2017. [Online]. Available: <http://bundymuseum.org/site3/about/the-history/willard-bundy-bio/>. [Accessed: 15-Dec-2017].
- [2] Skymark.com. “Frederick W. Taylor: Master of Scientific Management”, 2017 [Online]. Available: <http://www.skymark.com/resources/leaders/taylor.asp>. [Accessed: 15-Dec-2017].
- [3] Ko.com.ua, “Time Tracking Software - borcy s pozhiratelyami vremeni” (in Russian), 2013. [Online]. Available: http://ko.com.ua/time_tracking_software_borcy_s_pozhiratelyami_vremeni_81133. [Accessed: 15-Dec-2017].
- [4] Timetracker.biz, “Time Tracker” (in Russian), 2010. [Online]. Available: http://www.timetracker.biz/timetracker_function.html. [Accessed: 15-Dec-2017].
- [5] Blog-ru.primaerp.com, “Desyat' prichin dlya ispol'zovaniya tajm-trekerov” (in Russian), 2015. [Online]. Available: http://blog-ru.primaerp.com/2015/08/blog-post_26.html. [Accessed: 15-Dec-2017].
- [6] Coolwebmasters.com, “Luchshie MacOS programmy dlya otslezhivaniya rabochego vremeni dlya frilansera” (in Russian), 2012. [Online]. Available: <http://www.coolwebmasters.com/project-management/3313-time-tracking-mac-apps.html>. [Accessed: 15-Dec-2017].
- [7] Frey.su, “Trekeriy vremeni dlya programmista” (in Russian), 2015. [Online]. Available: <http://frey.su/programmer-time-tracker/>. [Accessed: 15-Dec-2017].
- [8] Onlineprojects.ru, “Time trackers” (in Russian), 2016. [Online]. Available: http://www.onlineprojects.ru/tags/tajm_trekeriy. [Accessed: 15-Dec-2017].
- [9] Lifehacker.ru, “5 trekerov vremeni, kotorye pomogut pobedit' prokrastinaciyu” (in Russian), 2016. [Online]. Available: <https://lifehacker.ru/2016/11/17/trekeriy-vremeni>. [Accessed: 15-Dec-2017].
- [10] IEEE Std 830-1998. IEEE Recommended Practice for Software Requirements Specifications. The Institute of Electrical and Electronics Engineers, Inc., 1998. ISBN 0-7381-0332-2.
- [11] Dpgrup.ru, “Trebovaniya k programmym produktam” (in Russian), 2017. [Online]. Available: <http://www.dpgrup.ru/software-requirements.htm>. [Accessed: 15-Dec-2017].
- [12] M. Miller, *Sams Teach Yourself...in 10 Minutes*. US: Pearson Education, 2014.
- [13] K. Adams, *Non-functional Requirements in Systems Analysis and Design*. US: Springer International Publishing, 2015.
- [14] S. Han, G. Skinner, V. Potdar, and E Chang, “A framework of authentication and authorization for e-health services,” in *Proc. 3rd ACM workshop on Secure web services*, Alexandria, Virginia, USA, 2006, pp. 105–106.
- [15] Electronjs.org, “Build cross platform desktop apps with JavaScript, HTML, and CSS”, 2017. [Online]. Available: <https://electronjs.org>. [Accessed: 15-Dec-2017].
- [16] T. Brown, *Web Development with Node and Express*. US: O'Reilly Media, 2014.
- [17] V.V. Gribova, and A.S. Kleshchev, “Metody i sredstva razrabotki pol'zovatel'skogo interfejsa: sovremennoe sostoyanie,” *Software & Systems*, vol. 1, pp. 1–6, 2001.
- [18] Smartbear.com, “Behind Every Piece of Quality Software is a Great Software Team”, 2017. [Online]. Available: <https://smartbear.com>. [Accessed: 15-Dec-2017].