EasyChair Preprint

№ 11511

A Proposal for Verbalization Data Gathering to Support Usability Evaluations with ErgoSV Tool

Cassilene Rodrigues de Assis, Tiago Coleti and Marcelo Morandini

December 12, 2023

A Proposal for verbalization data gathering to support usability evaluations with ErgoSV tool

1st Cassilene Assis
Universidade de São Paulo - USP
São Paulo, Brasil
cassia$_r p@yahoo.com.br$

2nd Tiago Coleti
Universidade Estatual do Norte do Parana
Parana, Brasil
thiago.coleti@uenp.edu.br

3rd Marcelo Morandini
Universidade de São Paulo - USP
São Paulo, Brasil
m.morandini@usp.br

***Abstract*—This paper aims to present a strategy to support the verbalization data gathering with ErgoSV tool to support usability evaluation. Verbalization is is a widely used and efficient based on the Think Aloud, once it provide resource to gather great amount of data about user interaction/experience. To support verbalization in usability testing, ErgoSV tool was proposed, but it was focused on a real time word recognition, which leaded to difficulties and limitations to conduct an appropriate speech recognition. Aiming to improve the verbalization data gathering, we propose a strategy where the recognition is conducted after data registration, avoiding processing overload and changing speech recognition techniques. The validation This research had used the practical research method, that has an applied nature. The method uses was quantitative analysis, with data obtained through field research based on observations. Through sample data, two types of users behaviors were identified: (1) in which the externalization differs from the questionnaire responses, and (2) in which the opinion coincides.**

***Index Terms*—usability, usability evaluation, think aloud, ErgoSV, user observation, questionnaires, usability tests**

I. Introduction

There are several interactive products inserted in people's daily lives, such as smartphone, computers and ATMs. However, some of these do not provide appropriated usability features. Usability is the main feature of interactive systems and refers to the ability which software allow their users to perform their tasks with effectiveness, efficiency and satisfaction [9].

Considering the software development, usability is a fundamental feature that aims to guarantee the better use of the system, since the demand of increasingly qualified users has been growing with the advancement of technology [15]. It also must be considered that not all users may have the needed experience, which places even more emphasis on the need on good levels of usability by designing products that present interactions opportunities with eficiency and effectiveness[17].

This paper aims to present a research that, by providing improvements in verbalization data gathering of the ErgoSV tool [2], aimed to support Usability Evaluation activities. This tool works based on the *Think Aloud* [15] technique in order to collect data about user opinion while he/she interacts with an application.

In order to provide the proposed improvements, changes were deployed in the the original version, which change the data collection strategy from a real-time strategy to asynchronous processing. With a asynchronous speech recognition, the tool became more efficient to support usability testing.

In this sense, we intend that: (1) data gathering can be conducted by any other tool, since it provide an audio file to be processed by new ErgoSV; (2) ErgoSV focus may be data processing and analysis heading data collect process to be performed by different tools.

To validate the new version of the ErgoSV, we conducted tests using audio sample data from usability evaluation. These audio were converted into texts to be analyzed how effective it were in recognition tasks, considering the word and the moment it was pronounced.

The next section introduces the Background. Next, section III Material and Methods. Section IV presents the Results. In section V Discussion and implications of the study, and section VI presents Conclusions

II. Background

This section presents the background about usability evaluation and the related works.

A. Usability Evaluation

According to Nielsen [15], "*usability is not a unique and uni-dimensional property of a user interface*", it is an attribute of the quality of the systems that aims to support user interaction and learning. Also, usability is defined by [8] as the degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

In the context of usability, evaluating methods aims to ensure a satisfactory and reliable level of usability. A widely method to evaluate software usability is the usability testing, which involve the real users interacting with the evaluated tool to perform real tasks [18]. Usability testing is considered appropriated method to measure the users interaction.

Usability testing may support user experience and interface projects, providing information to improve interfaces' quality and avoiding interaction errors. The data collected during these evaluations can be qualitative (descriptions that qualify usability) or quantitative (measures that quantify it) [14].

An usability testing technique is known as Verbalization (or Think Aloud). It is an observation method that depends on the direct participation of the user who must pronounce

what he/she is thinking while performing a task in order to externalize his/her reasoning processes. It is possible to carry out such observations in several ways and the records can be made through written notes, filming, or voice recorder. Obtaining such direct data is essential for identifying errors and possible causes, as well as possibilities for improvement. [10]. It is worth mentioning that recording using a voice recorder has the advantage of recording all the exact steps performed and explained aloud.

Regarding to qualitative data, it provide information based on narratives, ideas and individual experiences of the participants, having as main demand the satisfaction of the users' needs, which can be represented explicitly and/or implicitly. The data obtained from the observations during the evaluation activity can be compared to the expected usability standards. The quantitative usability evaluation is based on numbers, metrics and mathematical calculations that aim to obtain data through an evaluation that provides objective answers [21], which have the indicators of efficiency, effectiveness, satisfaction, intuitiveness, ease to learn, easy to remember [14].

Next section presents the related works.

B. Related Works

This section presents the related works and the contribution of this research considering them.

The main objective of usability evaluation support tools that uses video / audio recording as basis is to support the automate, synchronize and streamline the process of consolidating data collected through the recording of video, audio or screens. This strategy is considered time-consuming [4]. With the automation of evaluation tasks, it is possible to obtain a significant reduction in cost and complexity, in addition to eliminating specialists from repetitive and tedious tasks such as manual analysis of [16] logs.

There are several tools that support usability evaluations in desktop or mobile environments. Among them, this work highlights three of them, which have stood out in their applications:

- **IBM Rational Test Workbench [7]** is a platform composed of tools used to create, manage and run tests on the user interface of HTML 5 based web applications on desktop and mobile devices;
- **ErgoLight [5]** is a tool for collecting data on the behavior of real users under real operating conditions, aims to collect objective measures of user experience and operating efficiency, performs an analysis of the user and the barriers to continuous interaction and recommends possible design changes Based on actual usage data. ErgoLight consists of two modules:
  - *LogTester [12]*, is used to extract navigation diagnostics of websites from server log files;
  - *Lab-Tester [11]*, is used to detect and classify unexpected user events, for use in usability labs.
- **Loop11 [13]** is a tool to evaluate usability, it is composed of segmented *Feedback* modules, which provide resources for the elaboration of specific tests according to the websites to be evaluated. Specifically, the "Online Usability Testing" module, enables users to perform their tests online, and at the end, provides a result of understanding the user's behavior of HOW and WHY the website is being used.

In this section, we presented 3 tools that are specific for supporting the usability evaluation of websites and applications. Next section presents the ErgoSV, a tool that aims to support usability evaluations based on the *Think Aloud* technique, which focuses on the Evaluator's emotions at the moment they occur.

C. ErgoSV Tool

The ErgoSV tool was proposed by [2] and aimed to support usability evaluation tests using verbalization and filming strategies.

Through the *Think Aloud* method based on verbalization, this tool collects the words uttered during the tests, thus generating a set of raw data. Then, this data are processed and transformed into treated information to be used by those responsible for the evaluations and researchers. For the experiment, the Verbalization method relied on a list of keywords carefully chosen so that they were considered simple for the participants to memorize, pronounce and associate them with the interface resources. This aims to represent the evaluator's opinion about the interface evaluated at the time of the pronunciations. [2] concluded that the research approach was adequate, the data collection process was simple and fast for the evaluator and the person responsible for the evaluation, as well as the validation activities showed improvements in the time of data analysis and decision making.

III. Material and Methods

In this research, the audio of the participant's voice is recorded and converted into text. Several templates can be used to transcribe audio to text [6]. In particular, the SpeechRecognition[20] package, a speech recognition library in the Google, was used to perform this function. And Python [19] language was used to implement the adjustments.

A. SpeechRecognition

SpeechRecognition is a Google library for speech recognition that supports several *engines* and APIs[1], online or offline. In this research, *Google SpeechRecognition*[2] was used with a synchronous recognition request, which allows integration with the speech recognition technology of *Google API-Speech-to-Text*[3], in which its operation consists of receiving data audio, then process and recognize all of the audio, and return a text transcript in response. It is worth

[1]API:https://www.ibm.com/topics/api

[2]Google SpeechRecognition:https://pypi.org/project/SpeechRecognition/

[3]Google-API-Speech-to-Text:https://cloud.google.com/speech-to-text/docs/basics

noting that, as these are synchronous requests, the request method is blocked until *API-Speech-to-Text* responds to the previous request. The command lines below show the:

- audio capture
  audio = listen(source, phrase_time_limit=5, timeout=5)
  f.write(audio.get_wav_data())
  # phrase_time_limit = time in seconds that the recording lasts
  # timeout = time in seconds it takes to wait for the speech
  # get_wav_data = records captured audio to WAV file

- transcription to text
  audio = r.record(source) # reads the entire audio file
  texto_audio = r.recognize_google(audio, language='pt-br')
  # convert audio to text in brazilian portuguese language

In order to use all the functionalities of the *SpeechRecognition* library, the following requirements are necessary:

- Python 2.6, 2.7, or 3.3+
- PyAudio 0.2.11+ (since microphone input was used)

During the implementation process the following problems were encountered:

- Noise control: the high level of external noises can make the *recognizer* keep trying to capture and recognize speech even when no one is talking. This problem was minimized by calibrating the *recognizer* sensitivity to higher values, and thus making *recognizer* less sensitive. A point to be aware of is that the values to be calibrated will depend on the microphone or audio data used. There is no standard value, but it is possible to indicate that good values range from 50 to 5000 decibels. In this research, the calibrated value was 5000 decibels, as shown in the command line below.
  r.adjust_for_ambient_noise(source, duration=2) #duration = time in seconds it takes to analyze the audio source
- *Recognizer* initialization: when calibrated to very high values, it is possible that when starting *recognizer*, it does not recognize speech, because the *energy_threshold*[4] is being adjusted down automatically by the dynamic energy threshold adjustment, before being at a good level, the energy threshold is so high that speech is considered just ambient noise. It is possible to decrease the calibration value using the *energy_threshold* property, or use the *adjust_for_ambient_noise*[5] in advance, which will set the threshold to a good value automatically, as shown in the command line below.
  speech_recognition.Recognizer().adjust_for_ambient_noise (source, duration = 2) #duration = time in seconds it takes to analyze the audio source
- Power control: means the power level threshold for sound recognition. That is, the energy level for sounds considered silence or speech. The command lines below show the calibrated values for this search.
  energy_threshold = 5000 # values below 300 are considered silent and values above 300 are considered speech
  dynamic_energy_threshold = True # automatically adjusts sounds based on the current ambient noise level while listening
- Language: *recognizer* has English as the default language. For this research, it was necessary to set the language to Brazilian Portuguese, as shown in the command line below:
  speech_recognition.Recognizer().recognize_google(audio, language = 'pt-br')

B. PyAudio

PyAudio[6] is a library used to play and record audio from a microphone input. It was used in this research to communicate between the SpeechRecognizer and the microphone drive. Once installed, PyAudio is used in its standard form, it doesn't need any extra configuration, it is automatically recognized. The command lines below show how PyAudio was used in this research:
pip install pyaudio # library installation
mic = mr.Microphone() # library instance

C. Implementation

A adaptive maintenance of the ErgoSV tool was also part of the activities that involved this research. The main maintenance applied was specifically focused on the functionality of capturing the evaluator's verbalization while performing usability tests. For this purpose, the implementation was guided by the following scope:

- Replacement of *API* Coruja[7] used for the recognition and processing of words, by the SpeechRecognition library of the Python language;
- Construction of an audio to text converter module containing the following functionalities,
  - capture of words pronounced by the evaluator and recording in .WAV file,
  - transcription of audio streams to text using *Google-API-Speech-to-Text*
  - and recording of texts and matching words in the database and data;
- Construction of a word comparision module, which consists of comparing the words obtained by the audio to text converter module with the predefined keywords in the database.
- Construction of a Questionnaire module, which consists of registering the Questionnaires in the database, *frontend* for querying the Questionnaires and *frontend* of the questionnaire forms for registering them in the database.

[4]energy_threshold property:https://github.com/Uberi/speech_recognition/blob/master /reference/libraryreference.rst

[5]adjust_for_ambient_noise:https://github.com/Uberi/speech_recognition/blob/master /reference/libraryreference.rst

[6]PyAudio:https://pypi.org/project/PyAudio/

[7]Coruja, 2012. Software for voice recognition in Brazilian Portuguese, the website is not available for access.

- Construction of a report and results module, which consists of generating reports of the information obtained during the evaluation tests and the results obtained through comparisons of data from the audio versus responses from the Questionnaires.

Figure 1 presents how the ErgoSV tool is defined after these adjustments.

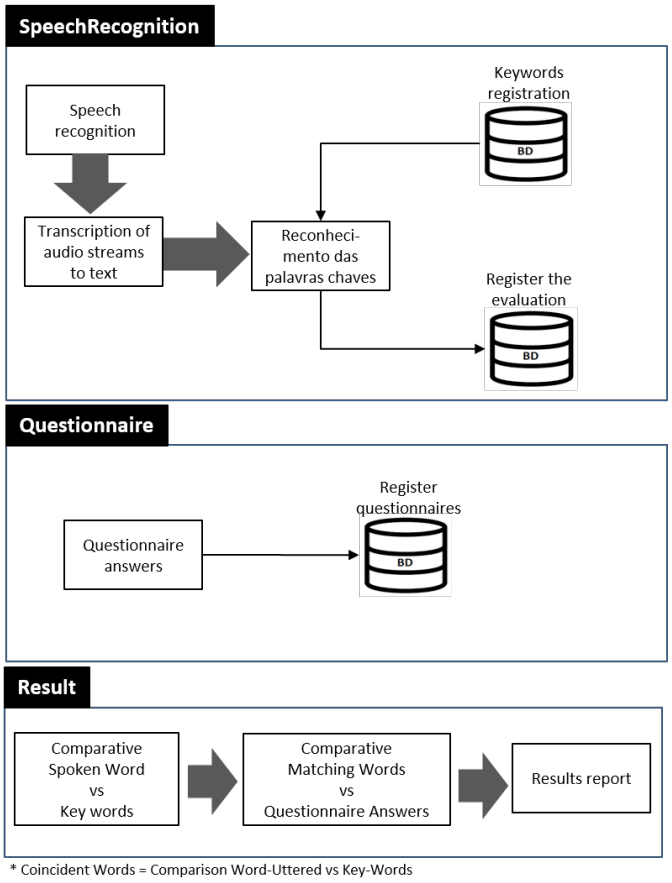

Fig. 1. Macro operation of the ErgoSV tool

D. Testing

With the adjusted version of the ErgoSV tool, it was possible to carry out the usability evaluation tests and generate sample data from the externalization and from the answers to the Questionnaires that provided the basis for the results analyzed in this research. Next is the sequence of carrying out the tests.

- ErgoSV Tool Configuration;
  - Registration of Keywords, which will be used in the comparisions of the text verbalized by the Evaluator. Each keyword will have a weight (score) and classification if Negative or Positive;
  - Registration of the duration of the tests, used for the audio evaluation.
  - Registration of the person responsible for the evaluation who will conduct the entire evaluation;
  - Registration of the Evaluator who will carry out the evaluation;
  - Registration of the Questionnaire to be answered by the Evaluator.
- Selection of website to be evaluated;
- Audio Evaluation - Audio Capture
  With an audio to text converter module active/on, the evaluation of the website selected starts, with the Evaluator being told to task of externalize its experience in the form of words and, specefically, keywords. During the entire time stipulated for the audio evaluation, every word said by the Evaluator is captured and recorded in WAV files. As in this research, the tests were controlled, the first author of this paper, herself, acted as Evaluator. In this case, the presence of a person responsible for the evaluation was not necessary to stimulate the externalization of words and keywords.
  For each evaluation, a directory is created and this directory name is the evaluation ID.
  In each directory, WAV files of approximately 5 seconds each, are recorded. To be used to define exactly at what time some keyword was pronounced, we defined as a list of .WAV files, in which each name of the file represents the position of the list where it was recorded..
- Evaluation Using the Questionnaires - Answering Questionnaire
  In this research, the technique of applying post-test Questionnaires was used after the interaction was concluded. It is a proposal that the evaluator might answer a questionnaire aiming at evaluating his/her satisfaction with the test performed. For the data sample generated in this research, the SUS questionnaire model was used (*System Usability Scale*) created by John Brooke [1], that can be used to evaluate products, services, *hardware*, *software*, websites, applications. The SUS consists of 10 questions, each of which the evaluator can answer on a scale of 1 to 5, where 1 means Completely Disagree and 5 means Completely Agree. Such questions aim to assess "Effectiveness" (Did the Evaluator complete its objectives?), "Efficiency" (how much effort and resources were required to do this?) and "Satisfaction" (was the experience satisfactory?).

In the context of this research, the "Capture Audio" and "Answer Questionnaire" processes are responsible for entering the primary data and initializing the data feed flow necessary for the analysis and generation of results.

IV. Results

From the sample data generated, the following results were obtained:

- Result of the audio collection - The converted text of the collected audio is visualized through Figure 2, which shows the result of crossing the keywords found in the text conversion, that is, 6 keywords were located in the audio converted into text.
  Next to the result of the identified keywords, the time interval in which the keyword was said is also informed.

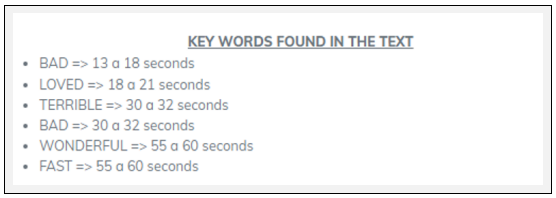

Fig. 2. Keywords found in converted text

- Result of the questionnaire collection - View of the Questionnaire Answers, in Figure 3 it is possible to visualize a answered questionnaire.

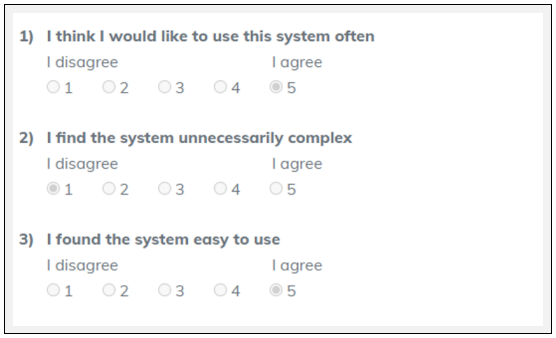

Fig. 3. Answered questionnaire

- Compararint the Results - ErgoSV has the functionality to compare the results of the audio evaluations and the answers to the Questionnaires, that is, the Comparison of Coincident Words vs Questionnaire Answers. The comparison is made from the score obtained by calculations of each one, which consists of:
  - Calculation of the score for evaluation by questionnaire, as already mentioned in this research, the SUS questionnaire model was used [1] (*System Usability Scale*), in that, for the odd answers (1, 3, 5), the value 1 is subtracted from the score that the Evaluator answered, for the even answers (2 and 4), the answer of 5 is subtracted, that is, if the Evaluator answered 2, then 5 – 2 = 3. Then all the values of the 10 questions are added together and multiplied by 2.5. And so, the final score will be calculated, which can range from 0 to 100.
  - Calculation of the score for the Coincident Words is done by separately adding the positive and negative coinciding words. Each word is registered with a score weight, as seen above in the ErgoSV configuration topic.

After performing the calculations mentioned above, the ErgoSV compares them and presents a suggestion about what was analyzed. The tool may conclude that the Evaluator had difficulties performing the test or that the Evaluator did not have difficulties performing the test, in which case the tool may suggest that a new test be carried out.

- Results comparison trend - The 20 sample evaluations generated in this research, were plotted in a line graph in which each point is the result of the comparison between the audio evaluation vs questionnaire responses, when the comparison is coincident, 2 is plotted and for the divergent ones, 1. A downward trend line is observed, which tells us that for this sample of 20 evaluations, the evaluators, when carrying out their evaluations, did not expose their true opinions. And this result is something that can be studied in future works. In Figure 4 it is possible to visualize the trend line of the 20 evaluations, commented in this topic

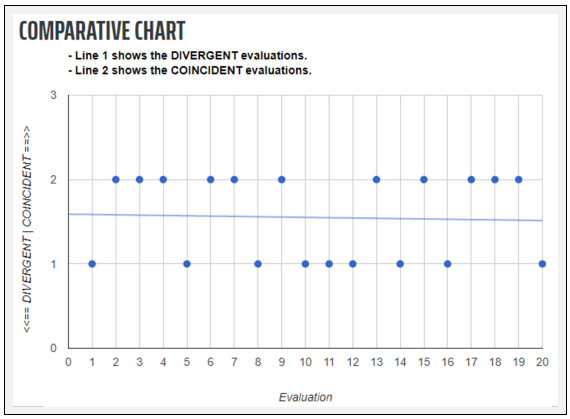

Fig. 4. Comparative chart

V. Discussion and Implications of the Study

For the user, the interaction can be considered from ”very easy to use” to ”extremely difficult to use”, and the difficulty in interaction is more easily noticeable as it brings specific constraints. Interfaces with a low level of adequacy to usability quality standards may become difficult for the user to learn and, so, he/she performs poorly, “leading” him/her to make mistakes and causing inconvenience and a feeling of incapacity [3]. In this context, validating the tests used to evaluate the usability of interactive interfaces is the main motivation of this research, which aims to observe the user during the execution of the usability evaluation tests supported by the ErgoSV tool, which is based on the *Think Aloud* technique for recognition of the Evaluator’s verbalization. About all aspects already mentioned, this research project seeks to answer the following question: What is the effectiveness of tests carried out to evaluate usability supported by the ErgoSV tool and by the application of Questionnaires? A scenario to be observed is that Evaluators may not be so sincere when responding to two different techniques on the same evaluation context, that is, when evaluating the usability of an interface first using verbalization and then answering a questionnaire, the Evaluator may pass two different results, therefore, it is possible to say that the application of two different ways to evaluate the usability that occurs in the same evaluation process will not be as effective, since the divergence between the answers

may imply in the final result, not leaving the Evaluator's real opinion is so clear, thus leading to the revalidation of the applied tests. It is worth mentioning that the reasons why the Evaluators may not be so sincere in their evaluations are also due to other factors that are not expected to be studied in this research.

VI. Conclusions

Evaluating the usability of the interfaces that make up the systems is a task that can demand time and resources. Thus, the use of tools to support the use of evaluation techniques, such as the ErgoSV, can be important to improve the quality of the process and, therefore, the final product developed. With the ErgoSV tool, it is possible to measure the usability of a product/system more assertively, regardless of the construction phase it is in, evolve test scripts that already exist or create new ones. The approach employed, which mixes the use of *Think Aloud* with the application of Questionnaires, proved to be significantly assertive and presents results that can measure the quality of the HCIs developed. This is due to the fact that the evaluator is given the freedom to express his emotions during the tests and when these emotions occur, making it possible to actually evaluate the interaction in terms of ease of learning, efficiency, effectiveness and satisfaction in use. In this context, another perspective is given to tests aimed at evaluating usability, that is, investing time to acquire and build customer loyalty. Capturing emotions, storing them in a database, enriching them by making them information that directs strategies and decision-making in corporations, makes the ErgoSV tool a strategic support tool, which does not remove its main objective from the search for a usable, fluid and pleasant product/system from the end user/client point of view. Comparative graphs of results obtained with the use of *Think Aloud* were presented together with the application of Questionnaires. Although only a few experiments have been carried out so far, it can be observed that this comparison presents indications of the quality of the evaluations themselves. The disagreement of some results of these evaluations may indicate that the evaluation process itself may have been impaired in some way. With that, the repetition of the evaluations by the specific evaluators can be recommended. A better prior training of them can also be suggested.

References

[1] John Brooke. "SUS: A quick and dirty usability scale". In: *Usability Eval. Ind.* 189 (Nov. 1995).

[2] Coleti T.A.; Morandini M.; Correa P.L.P.; Da Silva D.L.; Boscarioli C. "Using keywords to support the verbalization in usability evaluation". English. In: 2015. ISBN: 9781450353625.

[3] Adriana Holtz; FAUST Richard CYBIS Walter de Abreu; BETIO. *Ergonomia e usabilidade: conhecimentos métodos e aplicações*. Potuguese. Novatec, 2015. ISBN: 978-85-7522-459-5.

[4] Alan Dix et al. *Human-computer interaction*. Pearson Prentice Hall, 2004. ISBN: 0130461091.

[5] *ErgoLight*. http://ergolight.har-el.com/CHI/Human-Factors.html.

[6] AndeRson Luís FURLAN. "Desenvolvimento de um protótipo de aplicativo móvel para conversão de voz em texto e texto em voz, orientado ao apoio à comunicação de deficientes auditivos". PhD thesis. Universidade Federal de Santa Catarina, 2016. URL: https://repositorio.ufsc.br/bitstream/handle/123456789/176657/345870.pdf?sequence=1&isAllowed=y.

[7] *IBM Rational Test Workbench*. https://www.ibm.com/products/rational-test-workbench.

[8] ISO-25010. *ISO 25010: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. 2011. URL: https://www.iso.org/standard/35733.html.

[9] ISO-9241-110. *ISO 9241-110:2020(E) - Ergonomics of human-system interaction - Part 110: Interaction principles*. 2020. URL: https://www.iso.org/obp/ui/#iso:std:iso:9241:-110:ed-2:v1:en.

[10] V. Ade JESUS. "Atributos de Usabilidade para páginas web". PhD thesis. Faculdade de Jaguariúna, Jaguariúna, 2006.

[11] *Lab-Tester*. http://ergolight.har-el.com/CHI/Usage-Testing/Tools/Lab-Tester.html.

[12] *LogTester*. http://ergolight.har-el.com/CHI/Usage-Testing/Tools/Log-Tester.html.

[13] *Loop11*. https://www.loop11.com/.

[14] Marcelo MORANDINI. "Ergo-Monitor: monitoramento da usabilidade em ambiente web por meio da análise de arquivos de log". PhD thesis. Universidade Federal de Santa Catarin, 2003.

[15] J. NIELSEN. *Usability Engineering*. English. 1993. ISBN: 978-0-12-518406-9.

[16] Laila Paganelli and Fabio Paternò. "Intelligent analysis of user interactions with web applications". In: Jan. 2002, pp. 111–118. DOI: 10.1145/502716.502735.

[17] Morandini Marcelo; Coleti Thiago; Oliveira Edson; Corrêa Pedro. "Considerations about the efficiency and sufficiency of the utilization of the Scrum methodology: A survey for analyzing results for development teams". In: *Computer Science Review* 39 (Feb. 2021). DOI: 10.1016/j.cosrev.2020.100314.

[18] Jennifer Preece, Yvonne Borges, and Helen Sharp. *Design de Interação, Além da interação homem computador*. Bookman, 2005.

[19] *Python*. https://www.python.org/about/apps/.

[20] *SpeechRecognition*. https://pypi.org/project/SpeechRecognition/.

[21] Jacques Wainer. "Métodos de pesquisa quantitativa e qualitativa para a Ciência da Computação". In: (Nov. 2021).