Open Source Project Management Tools Assessment using QSOS Methodology

Anabela Carreira[1] and Jorge Bernardino[1,2][a]

[1]Polytechnic of Coimbra – ISEC, Rua Pedro Nunes, Quinta da Nora, 3030-199 Coimbra, Portugal
[2]CISUC - Centre of Informatics and Systems of University of Coimbra, Pinhal de Marrocos, 3030-290 Coimbra, Portugal

Keywords: Open Source Software, Project Management, Methodologies for Evaluating Open Source Software, Software Open Source, OSMM, QSOS, OpenBQR.

Abstract: With the increasing expansion of the open source tools in our daily life it is crucial to realize which are the best tools of the immensity that exist. In order to compare open source management tools, it is recommended to use a methodology such as QSOS that help to evaluate and choose which tool best suits our objectives. This paper describes some of the most commonly used open source project management tools such as GanttProject, OpenProject, and ProjectLibre and then compares them using QSOS methodology.

1 INTRODUCTION

Many of the projects nowadays continue to fail either because they are not delivered within the set deadline, or because they are not within the stipulated budget. The main reason for these problems is that "best practices" for project management continue to be adopted, as the project management knowledge guide PMBOK suggests (PMBOK, 2013).

The adoption of a project management tool can be an alternative to help with the integration of the model, thus allowing automation of the steps, thus increasing the efficiency of the process.

Currently, open source tools are increasingly present in our daily lives, whether they are free or because they also have high quality software (Laila et al., 2016).

The evolution of project management tools for software applications has been accelerating at a rapid pace and the number of open source project management tools available on the market has grown significantly.

Given this increasing expansion of open source software one of the main concerns is to understand the best methodologies to classify and evaluate this software and to understand how to apply the methodology and a project management tool.

This paper intends to perform a comparative ana-lysis between some of the most used open source tools in project management such as the GanttProject, OpenProject, and ProjectLibre tools. We use the QSOS methodology that allows this comparison to be made by assigning evaluations to each tool.

The rest of this paper is structured as follows. Section 2 describes each one of the tools, explaining what the functionalities, strengths and weaknesses of each one. Section 3 explaining the QSOS methodology, all its phases and how it should be applied in the context of tools for project management. Finally, section 4 presents the conclusions of the best tool for carrying out project management of the three tools used and the future work.

2 DESCRIPTION OF PROJECT MANAGEMENT TOOLS

In this section, we will perform a brief description of three popular open source project management tools: GanttProject, OpenProject, and ProjectLibre.

2.1 GanttProject

This software, which was developed by Alexandre Thomas at the University of Marne-la-Vallée

---

[a] https://orcid.org/0000-0001-9660-2011

(France), and is currently managed by Dmitry Barashev. GanttProject has been developed in Java and allows to run in different platforms, such as Windows, Linux and MacOS (GanttProject, GanttProject, 2019)).

The first version was launched on September 15, 2009. Its main features allow an access in order to manage the organization of the tasks, time and resources throughout the project. They also have a hierarchy between the various tasks and their dependencies, generate Gantt graphs, analysis and import reports, and export projects (Oliveira, 2013).

The main advantages of this tool are the following: allow we to send email directly to people (resource) involved in the project, indicate the critical path, allowing days off in the resources, indicate when an asset is overloaded or otherwise and also establish subtasks.

The negative aspects are not allowing naming "material" resources, that is, the necessary raw material, does not allow to enter costs to obtain the project budget, does not allow to change the period of the work, does not allow to define periodic tasks and still does not allow to change the period of work (Margea & Margea, 2011).

This tool has six tabs: one to control project, another to edit the project, one for the management of resources and another for the management of tasks, finally still has a tab that enables analysis of PERT.
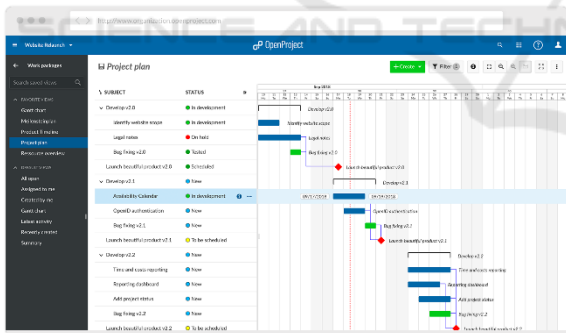

Figure 1: Interface of GanttProject (Source: (GanttProject, GanttProject, 2019)).

2.2 OpenProject

OpenProject has been developed since 2010, its first launch in October 2012 and credited by GNU (General Public). This project management tool is written in Ruby and Angular (OpenProject, 2019). This tool is translated into French, Spanish, German, Portuguese, Swedish, Finnish, Galician, Persian, Russian, Korean and is used in more than 142 countries. Its main functionalities are tasks, time, costs, the team and their frames.

Its main strengths are: schedules to follow the project in one of the supported methodologies, allows the Microsoft Project Native or Spring files without lose the files of the other application, wecan have the duration of tasks in days / weeks / years, subtasks, allows carry out a project budget and get the critical.

The main weaknesses are only to allow delay times to tasks that are in the calendar time, does not allow we to define tasks does not allow us to have a perception of the that is planned and the real one and does not allow resources automatically to resolve the problems of resource overload. This tool also has a Cloud version Edition and another Enterprise Edition (OpenProject, 2019).

This tool has several tabs: work packages, backlogs, calendar, wiki, cost reports, members, meetings, budgets and even project settings.
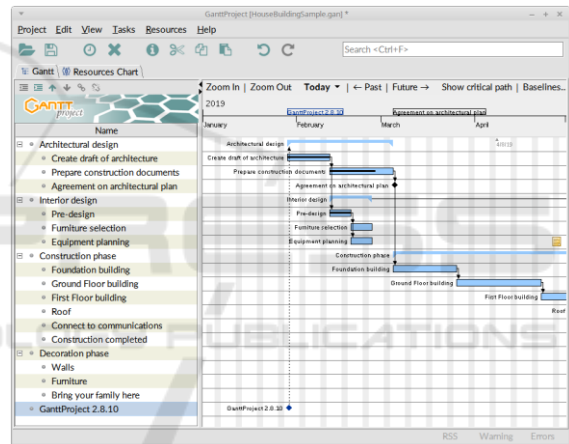

Figure 2: Interface of OpenProject (Source: https://www.openproject.org/).

2.3 ProjectLibre

ProjectLibre was created in Silicon Valley (France)by Marc O'Brien and Laurent Chretienneau, had the its first launch in August 2012. This tool was written in Java and is compatible with operating systems such as: Linux, Mac OS, and Windows (ProjectlLibre, 2019) .

This project management tool has emerged as an improvement latest released version of other management software OpenProj projects. This software allows we to perform the following functions: import / export files for Microsoft Project 2010, multiple users, generate analysis reports, perform a management of tasks, resources and costs.

The main points in favor of this tool are to be a free tool, be similar to Microsoft Project for anyone

who uses it and allow export / import files to Microsoft Project feature important for anyone who wants to migrate to this tool.

The less good points of this software are the fact that in this version it only has some of the features, it does not allow generating analytical reports or graphs. This application is only available in English language and has no translation in any language.

This tool has only four tabs: one to control the files, another to perform a resource management and another to manage the tasks, finally has a tab that allows we to view the views.
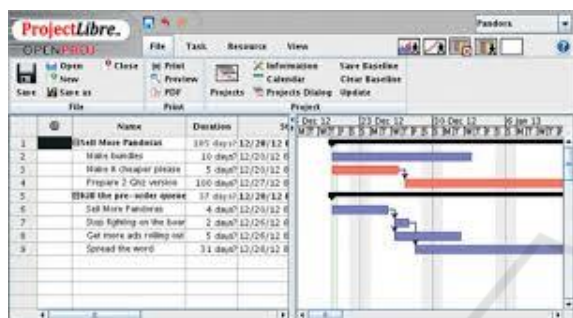

Figure 3: Interface of ProjectLibre (Source: www.projectlibre.com).

3 METHODOLOGY FOR SOFTWARE OPEN SOURCE EVALUATION

There are a variety of methods to evaluate open source software, based essentially on how they organize the project, their durability, maturity and some still add the functional aspects of the process (Pretinja, Sillitty & Succi, 2010).

The Open Source Quality Model (OSMM), Open Source Qualification and Selection of Open Source Software (QSOS) and the Open Business Quality Rating (OpenBQR) are the most commonly used methodologies (Laila et al., 2016).

Table 1: Comparison of methodologies.

Methodology	Evaluation Model	Punctuation Model	Year
OSMM	Scientific (3 levels)	1 a 5	2008
QSOS	Practical (3 or more levels)	0 a 2	2004
OpenBQR	Practical (3 levels)	1 a 5	2007

According to Table 1 it is possible to conclude that the QSOS is the method with the highest durability and with the highest levels of evaluation, although it only contains three types of punctuation.

3.1 QSOS

The QSOS methodology was launched in February 2012 by Atos. This method is intended to process and the evolution of open source software form to be assigned a score in each tool. QSOS consists of 4 iterative steps: Define, Evaluate, Qualify and Select.

Figure 4: Steps of QSOS methodology (Source: (Roumi, Spread your wings with free software, 2014)).

In the first step, the elements referring to the software typology are defined. This step is subdivided into 3 types: software type, license type and community type.

The type of software corresponds to the hierarchical classification of the software functional coverage grouped by the analysis of the project's maturity in relation to the changes occurred in the project. The type of license is classified into three categories:

Copyleft: are the most elaborate license when they are on the same owner or the same conditions.

Virality: when the software derives from one or more sources, and there is incompatibility with the licenses.

Inheritance: when it is possible for the license to be inherent in the group. As for the type of community, it can be classified as individual, group, commercial, etc.

Table 2: Scoring for the coverage criterion of software functionality.

Score	Criterion
0	Did not reach functionality
1	Functionality partially achieved
2	Fully achieved functionality

The second stage of this method corresponds to the evolution of the software, in this step a score is attributed to the software found in previous step.

These scores are assigned according to Table 2.

In step 3 that corresponds to the qualification the main objective is to define a set of requirements that match the needs in the context in which the tool will be used, thus generating a filter of the selection made. The first step of this step corresponds to identifying the filter that is intended for that specific tool. Subsequently the requirements for its maturity are classified, that is, if it is not relevant, relevant or critical. Finally, the requirements are evaluated for their functionality, therefore, they may be of the type required functionally, not required or optional (Galo Ramos, 2010).

The last step is to select the software that best suits your needs and compare with similar software. There are two approaches that can be used, one in which a rigorous selection is made in the other and a flexible selection is used (Souza, 2010).

Strict selection presupposes that when there is a requirement that does not meet the specifications, either the level of maturity, the desired functionalities or the generated filter, it must be eliminated.

Loose selection is less demanding than the previous selection because it does not eliminate the requirements that do not meet the specifications, classifies them, and measures their difference to the previously defined filters (Atos, 2013).

$$\bar{x} = \frac{\sum_{i=1}^{n} S_i * W_i}{\sum_{i=1}^{n} W_i}$$

This classification is made using the following equation:

Relative to weights for each criterion, either maturity or the functionality of the software.

Table 3: Weight for the maturity criterion of the software.

Weight	Description
0	Criterion not relevant (not included in filter)
1	Relevant criterion
3	Critical criterion

Table 4: Weight for the function criterion of the software.

Weight	Description
0	No functionality required
1	Optional functionality
3	Funcionality required

The software of the same domain must be compared to each other, using the weight of the scores obtained in the previous steps.

This comparison is made according to QSOS that provides tools that allow to apply the methodology in a much faster and easier way (Nereu, 2017).
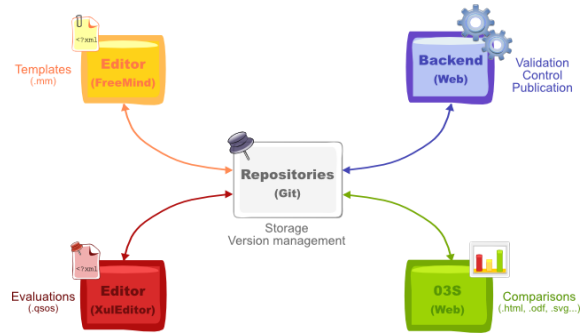

Figure 5: QSOS Method Tools and Formats. (Source: (Tools Use!, 2018)).

3.2 Applying the QSOS Methodology

In this section the QSOS methodology will be applied to the 3 project management tools.

Table 5: Information of each software.

Criterion	Gantt Project	Open Project	Project Libre
Company	The GanttProject Team	OpenProject Fundation	ProjectLibre Company
Creation	2003	2012	2012
Product	GanttProject	OpenProject	Project Libre
Lincense	GNU GPL v3	GNU v3	Common Public Attribution Licence
Version	2.8	8.0.0	1.7.1
Release	2016	2018	2017
Operating	Windows Linux Mac OS X	Cross-plataform	Linux Mac OS Windows
Community	https://help.ganttproject.biz/	https://community.openproject.com/	https://www.projectlibre.com/user-groups
Wiki	https://pt.wikipedia.org/wiki/GanttProject	https://www.openproject.org/help/wiki/	https://sourceforge.net/p/projectlibresdk/wiki/Home/
Forum	http://forum.ganttproject.biz/hot	https://www.openproject.org/help/forum/	https://sourceforge.net/p/projectlibre/discussion/
Url	https://www.ganttproject.biz/	https://www.openproject.org/	http://www.projectlibre.com/

In the first stage the sites of these applications were consulted in order to gather information regarding licenses, date of first launch, type of community, last version released, operating systems supported community sites, wiki, forum and also the download URL.

These results are presented in Table 5.

3.3 Results Obtained

In this section, we present the results obtained by applying the method to the different criteria.

The second step corresponds to the qualification, that is, grades between 0 and 2 are assigned based on the research performed for each of the applications in the previous step.

For each of the criteria a score of 0 to 2 was assigned, being that:
• 0 points corresponded to a system that did not support the criterion.
• 1 point, when the system supported much of the criterion.
• 2 points, when the system fully supported the criteria.

These notes are assigned according to the maturity table that this methodology shows.

Table 6: Notes attributed to the various maturity criteria.

Criterion	Gantt Project	Open Project	Project Libre
Age	2	2	2
History	2	2	1
Core team	2	2	1
Popularity	1	2	1
Contributing community	1	1	1
Activity on bugs	1	1	2
Activity on features	1	1	1
Activity on realeases	1	1	1
Copyright owners	1	1	1
Roadmap	1	1	1
Project management	1	1	1
Distribution mode	2	1	1
Services	2	1	1
Documentation	2	1	1
Quality assurance	2	1	1
Source core modification	1	1	1

According to the research carried out and the consultation of the official sites of the applications, each one of the tools was classified according to the functional criteria that this methodology proposes. This classification is made between 0 and 2, according to table 4 of the previous section.

Table 7: Notes attributed to the various functionality criteria.

Criterion	Gantt Project	Open Project	Project Libre
Full Stack	2	2	2
Enterprise-ready	2	2	1
Real-time Analytics	2	2	1
Solid and fault-tolerant	2	2	2
Scalability	2	1	2
Paid Version	2	1	1
User-friendly Management	2	1	2

In the 3 phase of this method a value between 0 and 3 is assigned according to the degree of relevance of the criterion for the study that is being carried out. These values are presented in Tables 8 and 9.

Table 8: Weight of maturity criteria.

Criterion	Weight
Age	3
History	3
Core team	0
Popularity	3
Contributing community	3
Activity on bugs	3
Activity on features	3
Activity on releases	2
Copyright owners	1
Roadmap	0
Project management	1
Distribution mode	1
Services	3
Documentation	3
Quality assurance	2
Source core modification	2
Total	**33**

Table 9: Weight of functionality criteria.

Criterion	Weight
Full Stack	3
Enterprise-ready	3
Real-time Analytics	1
Solid and fault-tolerant	2
Scalability	1
Paid Version	2
User-friendly Management	3
Total	**15**

Table 10: Total comparison of functionality criteria.

Criterion	Gantt Project			Open Project			Project Libre		
	S	W	R	S	W	R	S	W	R
Full Stack	2	3	6	2	3	6	2	3	6
Enterprise-ready	2	3	6	2	3	6	1	3	3
Real-time Analytics	2	1	2	2	1	2	1	1	1
Solid and fault-tolerant	2	2	4	2	2	4	2	2	4
Scalability	2	1	2	1	1	1	2	1	2
Paid Version	2	2	4	1	2	2	1	2	2
User-friendly Management	2	3	6	1	3	3	2	3	6
Total	30/15=2			24/15=1.6			24/15=1.6		

Table 11: Total comparison of maturity criteria.

Criterion	Gantt Project			Open Project			Project Libre		
	S	W	R	S	W	R	S	W	R
Age	2	3	6	2	3	6	2	3	6
History	2	3	6	2	3	6	1	3	3
Core team	2	0	0	2	0	0	1	0	0
Popularity	1	3	3	2	3	6	1	3	3
Contributing community	1	3	3	1	3	3	1	3	3
Activity on bugs	1	3	3	1	3	3	2	3	6
Activity on features	1	3	3	1	3	3	1	3	3
Activity on releases	1	2	2	1	2	2	1	2	2
Copyright owners	1	1	1	1	1	1	1	1	1
Roadmap	1	0	0	1	0	0	1	0	0
Project management	1	1	1	1	1	1	1	1	1
Distribution mode	2	1	2	1	1	1	1	1	1
Services	2	3	6	1	3	3	1	3	3
Documentation	2	3	6	1	3	3	1	3	3
Quality assurance	2	2	4	1	2	2	1	2	2
Source core modification	1	2	2	1	2	2	1	2	2
Total	48/33=1.45			42/33=1.36			39/33=1.18		

The selection phase corresponds to be selected with the use of this methodology best tool, for this, after being assigned weights for each criterion, the results are then calculated for each criterion. It starts by multiplying the score by the weight, thus obtaining a result for each criterion. At the end is a sum of all the criteria to calculate the average, it is only necessary to divide by the sum of the weight. In Table 9, the various scores and weights attributed to the functional criteria can be consulted and the best functional tool for project management is still concluded, on the other hand, in Table 10 it is possible to know the best tool in terms of maturity.

Table 12: QSOS Evaluation results.

Criterion	GanttProject	Open Project	Project Libre
Maturity	1.45	1.36	1.18
Functionality	2	1.6	1.6
Total	3.45	2.96	2.78

It is verified by the analysis of Table 11 that the application that obtained a better score in the several criteria evaluated in the previous tables (Tables 10 and 9) was the tool GanttProject, being the one that has better performance in terms of functionalities as well as of maturity.

4 CONCLUSIONS AND FUTURE WORK

The selection of an appropriate open source project management tool is a challenge task. Based on our assessment, the ´most advantageous tool was the GanttProject having obtained the score of 1.45 points in the criteria of maturity and 2 points in the functional criteria obtained an overall score of 3.45 values. In this way it was the one that presented better results in all the groups of the chosen criteria.

What was ranked second according to this methodology was OpenProject, with general average of 2.96 points. Finally, ProjectLibre with a general average of 2.78 points. It should be noted that OpenProject and ProjectLibre achieved equal performance in the functional criteria, but that OpenProject stood out slightly in the maturity criteria.

As future work, we intend to compare these three tools: GanttProject, ProjectLibre and OpenProject using other methodologies of comparison of open source tools to understand if the results concluded with the use of this method are maintained with another approach.

REFERENCES

Atos. (2013). *QSOS*. Obtido de QSOS: http://edoc.qsos.org/en/qsos-2.0_en#comparison Ferreira, M., Ferros, L. M., & Fernandes, V. (Outubro de 2012). Avaliação e seleção de software open-source para gestão integrada de bibliotecas.

Galo Ramos, J. P. (2010). anazlisis del meztodo para calificaciozn de software qsos para la selecciozn de software aplicable a procesos educativos. *selecciozn de software aplicable a procesos educativos*, pp. 103-123.

GanttProject. (2019). Obtido de wikipedia: https://en.wikipedia.org/wiki/GanttProject

GanttProject. (29 de Janeiro de 2019). *GanttProject*. Obtido de GanttProject: https://www.ganttproject.biz/

GanttProject. (2019). *GanttProject*. Obtido de https://www.google.com/search?q=interface+ganttproject&source=lnms&tbm=isch&sa=X&ved=0ahUKEwivp4apjeLiAhXF1eAKHeoYCZcQ_AUIECgB&biw=1366&bih=657#imgrc=JRi7BDIyl8464M:

Laila, U. e., Zahoor, A., Methboob, K., & Natha, S. (2016). Comparison of open source maturity models. *8th International Conference on Advances in Information Technology* (pp. 19-22). Macau,China: ScienceDirect.

Margea, R., & Margea, C. (1 de 2011). Open Source Approach to Project Management Tools. *Open Source Approach to Project Management Tools*, p. 196.

Nereu, J. F. (Outubro de 2017). Open Source Platforms for Big Data. *Open Source Platforms for Big Data*.

Oliveira, J. F. (2013). *Utilização de ferramentas Informáticas na gestão de Projets-enfoque na gestão colaborativa.* Minho.

OpenProject. (25 de setembro de 2018). Obtido de Wikipedia: https://en.wikipedia.org/wiki/OpenProject

OpenProject. (2019). *OpenProject*. Obtido de OpenProject: https://www.openproject.org/

PMBOK. (2013). *Guia PMBOK.* Pennsylvania: Project Management Institute.

Pretinja, E., Sillitty , A., & Succi, G. (2010). Comparing OpenBRR, QSOS, and OMM. *IFIP AICT- Comparing OpenBRR, QSOS, and OMM*, pp. 225-238.

ProjectLibre. (2019). Obtido de ProjectLibre: https://www.projectlibre.com/

ProjectlLibre. (2019). *ProjectlLibre*. Obtido de ProjectlLibre: www.projectlibre.com

Roumi, A. (2014). *Spread your wings with free software*. Obtido de Spread your wings with free software: https://amalroumi.wordpress.com/

Roumi, A. (s.d.). *Spread your wings with free software*. Obtido de Spread your wings with free software: https://amalroumi.wordpress.com/2013/11/30/open-source-software-assessment-methodologies/

Souza, R. S. (2010). Avaliação de Ferramentas CMS Ruby on Rails. *Avaliação de Ferramentas CMS Ruby on Rails*, pp. 102-124.

Tools Use! (2018). Obtido de QSOS: https://www.google.com/search?biw=1329&bih=627&tbm=isch&sa=1&ei=KIK3XJONJ5GalwTa9ZXADQ&q=qsos+tools&oq=qsos+tools&gs_l=img.3..35i39.2111130.2113539..211 3828...0.0..0.97.525.6......1....1..gws-wiz-img.rhPqwLEvA6g#imgrc=w9Ql9qchOt563M: