OSS popularity: understanding the relationship between user-developer interaction, market potential and development stage

Clémence Cheruy
Montpellier Business School, France, c.cheruy@montpellier-bs.com

Nassim Belbaly
Montpellier Business School, France, n.belbaly@montpellier-bs.com

Franck Robert
Montpellier Business School, France, f.robert@montpellier-bs.com

Follow this and additional works at: https://aisel.aisnet.org/sim

Recommended Citation
Cheruy, Clémence; Belbaly, Nassim; and Robert, Franck (2017) "OSS popularity: understanding the relationship between user-developer interaction, market potential and development stage," Systèmes d'Information et Management: Vol. 22 : Iss. 3 , Article 3. Available at: https://aisel.aisnet.org/sim/vol22/iss3/3

This material is brought to you by the AIS Affiliated and Chapter Journals at AIS Electronic Library (AISeL). It has been accepted for inclusion in Systèmes d'Information et Management by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
OSS popularity: 
Understanding the relationship 
between user-developer interaction, 
market potential 
and development stage

Clémence CHERUY*, Frank ROBERT** & Nassim BELBALY**

* Université Toulouse III Paul Sabatier, Laboratoire Gouvernance 
et Contrôle Organisationnel, France
** Montpellier Business School

ABSTRACT

Following the growing interest and concerns regarding open source software (OSS) phenomenon among academics and practitioners, many studies have been conducted to understand the factors that influence OSS success. However, research has primarily explored such factors in the context of well-known projects, such as Linux and Apache. Yet, lesser-known projects must be examined to gain a more complete understanding. Accordingly, this paper focuses on lesser-known projects to examine three factors that influence OSS popularity: user-developer interaction, market potential and development stage. Specifically, we develop an empirical model of OSS popularity and test our hypotheses on data from 657 open source projects. The findings show that the combination of the three factors has a positive effect on OSS popularity. Moreover, in contrast to previous research, the results reveal that exchanges among users and developers have a stronger influence on OSS popularity than bug-related activities. Overall, this research provides a novel way to measure OSS popularity for lesser-known projects and offers organizations a better understanding of OSS.

Keywords: Open source software, OSS development, OSS popularity, User-developer interaction, Market potential, Development stage.

RÉSUMÉ

Alors que le logiciel libre a particulièrement retenu l’attention et l’intérêt du monde académique et managérial, beaucoup de recherches ont cherché à comprendre les facteurs de succès de ces logiciels OSS. Cependant, ces recherches se sont principalement concentrées sur des projets reconnus et de grande envergure, tels que Linux ou Apache. Une investigation des projets moins connus aurait permis une compréhension plus complète et d’éviter un taux d’échec important des projets OSS. Cette étude explore donc la combinaison de trois facteurs pour comprendre l’op-
raison de trois facteurs : les interactions des utilisateurs et développeurs, la pénétration du marché et le stade de développement, afin d’observer leur impact sur la popularité des projets OSS peu connus. Nous avons pour cela développé un modèle empirique de la popularité des OSS, et avons testé nos hypothèses sur 657 projets de logiciel libre. Les résultats de cette recherche montrent que la combinaison des trois facteurs a un impact positif sur la popularité des OSS. De plus, contrairement aux études précédentes, nos résultats révèlent également que les interactions entre utilisateurs et développeurs ont un rôle plus important que les activités reliées à la résolution de bugs dans la réussite des projets. Cette recherche propose une nouvelle manière de mesurer la popularité d’un projet OSS et offre ainsi aux organisations une meilleure compréhension.

Mots-clés : Logiciel libre, développement logiciel, popularité du logiciel, interaction utilisateurs/développeurs, stage de développement.

1. INTRODUCTION

Although the Open Source Software (OSS) phenomenon has long been characterized by the collaboration of volunteers who supply their work for free (Benbya & Belbaly, 2010), this phenomenon has progressively evolved towards a more commercially viable form labelled OSS 2.0 (Fitzgerald, 2006). The notable success of key OSS projects, such as the Linux computer operating system and the Perl programming language, has increased academic interest in OSS (Von Hippel & Von Krogh, 2003; Crowston et al., 2012), and such interest has been nurtured by the need to understand the high failure rates of OSS projects to improve their overall success (Fitzgerald, 2009; Sutanto et al., 2014). Accordingly, numerous studies have focused on the determinants of OSS project success (Crowston et al., 2003, 2006; Fitzgerald, 2009; Midha & Palvia, 2012) by examining, for example, the influence of developers’ motivations (Stewart et al., 2005; Meissonier et al., 2010), user utility (Stewart et al., 2005), internal cohesion (Singh et al., 2011), developers’ technical achievements, and indicators of market success (Grewal et al., 2006) to better understand how OSS projects succeed.

The definition of OSS success has different meanings across projects and stakeholders. OSS success is a development-oriented measure and generally refers to the level of activity on a project (Stewart et al., 2005). However, this definition is highly disputed (Crowston et al., 2003) and somewhat difficult to measure. Therefore, some studies have proposed OSS popularity as a valuable indicator of the success of an OSS project, especially for lesser-known projects (Stewart & Ammeter, 2002; Crowston et al., 2012). However, research on the factors that influence OSS popularity remains limited, and the few existing studies mainly focus on user involvement (e.g., Von Hippel & Von Krogh, 2003), the support network (Capra et al., 2011; Sutanto et al., 2014), project-specific characteristics such as the project age, software user license or target users (Stewart & Ammeter, 2002), and network embeddedness (Grewal et al., 2006).

Consequently, further research is needed to examine OSS popularity by investigating other factors, or combinations of factors, to create new knowledge pertaining to open source projects (e.g., Von Hippel & Von Krogh, 2003; Singh et al., 2011). Evaluating OSS popularity based on the combined effects of market potential (Grewal et al., 2006), user-developer interaction (Wågstrom et al., 2005; Barcellini et al., 2008; Iivari, 2009a), and development stage (Stewart & Gosain, 2006) constitutes a different and logical way of determining the success of OSS projects. These
factors can benefit from the recognition of their joint effects on users, developers, project administrators and organizations. Further, a better understanding of the role of OSS actors would allow organizations to develop a favourable environment for stimulating and fostering OSS development. Although the three aforementioned factors have been studied separately, there is little evidence regarding their combined explanatory power for OSS success. Therefore, we believe that a joint focus on market potential, user-developer interaction, and development stage may be helpful to establish a better understanding of OSS popularity.

Though market penetration is often used to examine well-known projects such as Linux and Apache (Gallivan 2001; Mockus et al. 2002), the popularity of a project among potential users is a similar indicator that can be used for other OSS projects, including projects with much lower levels of participation and prominence, and is more representative of the majority of OSS projects (Krishnamurthy, 2002; Stewart & Gosain, 2006). Lesser-known projects are defined as projects that are not yet popular (renowned) in the community. Thus, by measuring the general level of interest in a project (i.e., its popularity measured by the number of downloads), we can better understand how OSS projects attract interest and input from the user-developer community. We can also provide a better way of assessing the overall success and interest level for OSS projects. Therefore, this study aims to explore lesser-known projects to obtain such evidence.

This research specifically investigates the following research questions: Does the combined effect of user-developer interaction and market potential lead to greater OSS popularity? Moreover, how does the development stage of a project influence this combined effect on OSS popularity? To answer our research questions, we tested our model by using data on 657 open source projects hosted by Sourceforge.net that were focused on enterprise application development. The results indicate that the combination of market potential, user-developer interaction and development stage has a positive impact on OSS popularity.

We proceed as follows. The second section of the paper presents the theoretical background, the first subsection of which is dedicated to previous research on OSS success, a second subsection in which we explain the notion of OSS popularity, and a third subsection in which we present our conceptual model and formulate hypotheses regarding the factors that affect OSS popularity. The third section then presents the methodology used to test the model and the data analysis, and the fourth section reports the results. The implications of the results for theory and practices are subsequently discussed in the fifth section. Finally, the paper concludes with the limitations of our research and avenues for future research.

2. THEORETICAL DEVELOPMENT

In our case, we will focus on open source using a crowdsourcing platform made for OSS development – Sourceforge.net. This platform integrates developers and user’s interactions with the aim of developing publicly available software. From a research perspective, this platform will support our understanding on how user-developer interactions, coupled with market-based measures and development stage variables, explain OSS popularity. Moreover, the aim of this paper is to explore OSS popularity to better understand its antecedents and determinants for lesser-known projects. However, the notions of popularity and success are often confused in the literature. Thus, we will first refer to the literature describing OSS success and its determinants to better
understand the notion of success and its related limitations. Then, we will develop the notion of OSS popularity, explaining why we prefer to use this performance metric in our study.

2.1. Key determinants of OSS success

The success of OSS projects has become a critical topic with the growing usage of OSS products by private and government organizations (Sen, 2007a). In their taxonomy of OSS research, Aksulu and Wade (2010) grouped together performance metrics and identified quality and success as the two major metrics used to measure OSS performance. Software quality generally refers to OSS features such as usability, feasibility, and adaptability and their impact on OSS adoption and diffusion (e.g., Jørgensen, 2001; Yu et al., 2006), whereas studies on OSS success generally focus on factors affecting or encouraging the value of OSS (Aksulu & Wade, 2010).

In our study, we focus on OSS success. A common method for examining OSS success is to link it to the IS success model developed by DeLone and McLean (1992, 2002, 2003), which includes six interrelated measures of success: system quality, information quality, use, user satisfaction, individual impact, and organizational impact. However, these measures do not always consider the unique characteristics of OSS, such as the development environment in which an OSS project evolves (Crowston et al., 2003). Moreover, the IS success model rely mainly on the adoption by both individual and organizational unit (e.g. Subramanian & Sen, 2009) and does not consider the success of OSS as a product (Crowston et al., 2006). In our case we are studying the product success of the OSS developed.

To overcome these issues, previous studies have analysed different factors affecting project success. To better understand these factors, we propose an ad hoc OSS classification system in which categories are used to summarize empirical observations related to OSS success (adapted from Webster & Watson, 2002). We proceeded as follows. First, we divided our primary concept, the determinants of OSS success, into two subcategories: internal and external factors. Internal determinants of OSS success are factors directly related to software itself, such as the type of license (e.g., Comino et al., 2007; Subramaniam et al., 2009) we investigate open source software (OSS, software quality (e.g., Lee et al., 2009), internal community (e.g., Meissonier et al., 2010), and OSS norms and values (e.g., Stewart & Gosain, 2006), and their potential impacts on OSS success. Because OSS success also depends on external relationships that project members have with developers outside of the focal project (Singh et al., 2011), external determinant factors refers to the impact of external interactions and interest level, such as the community service quality (e.g., Grewal et al., 2006; Lee et al., 2009), network (e.g., Dalhander & Magnusson, 2005; Méndez-Durón & Garcia, 2009), commercialization and sponsorship (e.g., Lerner & Tirole, 2005), and user interest (e.g., Bagozzi & Dholakia, 2006; Von Krogh & Von Hippel, 2006). The internal and external factors that we highlight in Table 1 have been demonstrated to be drivers of OSS success.

However, the definition of success has differed across projects and stakeholders (Crowston et al., 2003), and previous studies have commonly acknowledged that the success of OSS projects can be interpreted in different ways (Crowston & Scozzi, 2002; Capra et al., 2011). Grewal et al. (2006) explain that measuring the success of OSS projects in terms of technical achievements or market success represents an incomplete picture of success. Several studies used
the term popularity to examine the performance of OSS projects. Consequently, popularity can complete this picture as an indicator of OSS success. Hence, rather than explaining success of OSS, we focus on OSS popularity as an outcome of OSS performance. In the next section, we develop the notion of popularity.

<table>
<thead>
<tr>
<th>Articles</th>
<th>Determinants of OSS success</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Internal Factors</td>
</tr>
<tr>
<td></td>
<td>OSS type</td>
</tr>
<tr>
<td>Bagozzi &amp; Dholakia (2006)</td>
<td></td>
</tr>
<tr>
<td>Bonaccorsi &amp; Rossi (2003)</td>
<td></td>
</tr>
<tr>
<td>Brabham (2008)</td>
<td></td>
</tr>
<tr>
<td>Cai &amp; Zhu (2016)</td>
<td></td>
</tr>
<tr>
<td>Comino et al. (2007)</td>
<td>x</td>
</tr>
<tr>
<td>Crowston &amp; Scozzi (2002)</td>
<td>x</td>
</tr>
<tr>
<td>Dahlander &amp; Magnusson (2005)</td>
<td></td>
</tr>
<tr>
<td>Grewal et al. (2006)</td>
<td></td>
</tr>
<tr>
<td>Lee et al. (2009)</td>
<td></td>
</tr>
<tr>
<td>Lerner &amp; Tirole (2005)</td>
<td>x</td>
</tr>
<tr>
<td>Meissonier et al. (2010)</td>
<td></td>
</tr>
<tr>
<td>Méndez-Durón &amp; Garcia (2009)</td>
<td></td>
</tr>
<tr>
<td>Peng et al. (2015)</td>
<td></td>
</tr>
<tr>
<td>Sarker &amp; Schneider (2009)</td>
<td></td>
</tr>
<tr>
<td>Sen (2007a, b)</td>
<td></td>
</tr>
<tr>
<td>Sen et al. (2015)</td>
<td>x</td>
</tr>
<tr>
<td>Sen et al. (2007a), Sen et al. (2012)</td>
<td>x</td>
</tr>
<tr>
<td>Shah (2006)</td>
<td></td>
</tr>
<tr>
<td>Stewart &amp; Gosain (2006)</td>
<td></td>
</tr>
<tr>
<td>Stewart et al. (2006)</td>
<td>x</td>
</tr>
<tr>
<td>Subramaniam et al. (2009)</td>
<td>x</td>
</tr>
<tr>
<td>Von Krogh &amp; Von Hippel (2006)</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Antecedents of OSS success
2.2. The notion of OSS popularity

Stewart et al. (2005, 2006) found that popularity is related to one measure of OSS success and define it as the level of interest and attention that the project generates among current and potential users (Stewart et al., 2005; Crowston et al., 2012). This parameter is crucial to estimate a project potential, as it is an indicator of how the community (users and developer) assesses a project (Capra et al., 2011). The level of interest is commonly measured by indicators such as the number of downloads. This component is particularly relevant as it evaluates the number of unique downloads of a project, thus considering the popularity of the project from the final user’s perspective (Crowston et al., 2003; Capra et al., 2011). It allows decision-makers to revaluate, adjust, or rectify how to lead the project, directly reflecting the current level of interest (Stewart et al., 2005).

Thus, OSS popularity is considered as a key success factor for OSS projects, since it evaluates the extent to which an OSS project is able to attract community interest in the project software (Subramaniam et al., 2009; Ghapanchi, 2013). Moreover, it is much easier to gather data on popularity factors such as user and developer interest due to the openness of the development environment (Ghapanchi, 2013). As a result, academics started to investigate antecedents of OSS popularity to better understand why a project becomes successful or not. We summarize empirical observations related to OSS popularity using an ad hoc classification (adapted from Webster & Watson, 2002) of OSS success factors. We use external and internal factors as well to classify OSS popularity determinants (Table 2).

When comparing OSS success and OSS popularity factors, the first observation that we can make is that far fewer studies have investigated the notion of OSS popularity than those that have examined OSS

<table>
<thead>
<tr>
<th>Articles</th>
<th>Internal Factors</th>
<th>External Factors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OSS license</td>
<td>Firm involvement</td>
</tr>
<tr>
<td>Capra et al. (2011)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ghosh et al. (2013)</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Sen (2007a,b)</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Stewart &amp; Ammeter (2002)</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Stewart et al. (2005)</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Sutanto et al. (2014)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Antecedents of OSS Popularity
success factors. Second, and as argued by Ghapanchi (2015), a review of these prior studies has identified a lack of literature exploring software development process considerations that predict user interest in an OSS project (Conley, 2008). Third, Crowston et al. (2003) explain that the development environment in which an OSS project evolves must account for user interest. Thus, Crowston et al. (2003) identify three measures that are more representative of OSS project performance: project activity level, development/team community, and time required to fix bugs. More specifically, Crowston et al. (2003) argue that instead of measuring the actual use, the particularities of an OSS project must be measured by assessing input (level of activities), process (speed of bug fixing) and output (popularity). While previous research mainly explores isolated factors (cf. table 2) rather than exploring only the development process, which can explain popularity, we think that we need a deeper examination of the input, process and output as determinants of OSS popularity.

We identify that (1) success directly refers to the level of a project, whereas popularity refers to the level of individual attraction (e.g., Sutanto et al., 2014), and (2) less is known about the determinants of popularity (Crowston et al., 2003; Ghapanchi, 2015). To overcome these gaps in the literature, our study focuses on OSS inputs based on user-developer activities, the OSS development process, including the status of the project, and OSS popularity as an outcome. The next subsection explains the variables we use as drivers of popularity to develop our hypotheses and research model.

2.3. Hypotheses and research model development

To understand OSS project popularity, we must consider why users would choose one software solution over another (Stewart et al., 2005). As suggested by Crowston et al. (2003), we build our model by following the overall OSS development process, which includes input, process and output (figure 1).

![Figure 1. Development process of OSS project](image-url)
First, the input considers the relationship between the user and developers by using information flow, bugs and market-based measures. Indeed, successful OSS/technical support (OSS developer support) increases the fit of a project interest with OSS users, which can increase OSS popularity (Ghosh et al., 2013). Past research highlights that the OSS project environment enables more diversity and creativity given the high level of interactions and communication between different parts, producing higher levels of innovation (Vujovic & Ulhøi, 2008; Aksulu & Wade, 2010). These interactions and the resulting potential creativity are essential to attracting potential users, thus generating popularity (Nambisan & Wilemon, 2000). In addition, we must include both the developers’ technical achievements and indicators of market success to provide a comprehensive overview of OSS performance (Grewal et al., 2006). Thus, market potential can be examined as the right indicator of general project interest and market success (e.g., Grewal et al., 2006). OSS market potential can be explained by using network embeddedness theory (Sutanto et al., 2014), which suggests that potential users can be influenced by user support provided by the community regarding product use (voluntary contributions within an OSS community). Hence, when the number of queries increases, the number of visits to the OSS project website increases (Singh et al., 2011; Sutanto et al., 2014). Consequently, the market potential of the OSS project increases, positively affecting the popularity of the project. Second, to evaluate the notion of process, we consider the evolution of the project itself. Based on the IS literature, the project stage is likely to be a salient contingency factor that affects OSS project performance (e.g., Majchrzak et al., 2000; Qureshi & Vogel, 2001). Given the iterative process of an OSS project, which results from the interaction and re-adaptation between developers and users based on users’ needs, the development stage should also influence OSS popularity because more mature and stable OSS projects should have greater numbers of views and downloads. However, many prior studies have used the development stage only as a control variable and not as a factor that influences OSS popularity (Stewart & Gosain, 2006; Singh et al., 2011; Sutanto et al., 2014). Third, the output includes studies on OSS popularity, which have investigated the previous influencing factors separately; therefore, examining their combined effect will play a critical role in providing new knowledge regarding open source projects (e.g., Von Hippel & Von Krogh, 2003; Singh et al., 2011). Overall, the effect of the user community (e.g., Dahlander & Magnusson, 2005; Toral et al., 2010) on OSS popularity depends on the combined effects of user-developer interaction (Wägstrom et al., 2005; Barcellini et al., 2008; Iivari, 2009b); market potential, which signals the general interest level (Grewal et al., 2006); and development stage (Stewart & Gosain, 2006). Therefore, this paper investigates user-developer interaction, market potential and development stage as antecedents of OSS popularity (Figure 2).

2.3.1. User-developer interactions

To explain the role of user-developer interaction in the context of OSS development, we rely on the phenomenon of mutual adaptation in the technology transfer process (Leonard-Barton & Sinha, 1993). The process of transferring a technical system from developers to users always differs depending on the context (Leonard-Barton & Sinha, 1993). Thus, additional transformations and adaptations are required to fit the system to the operating environment. In other words, even if the developers fit their system to their original technical objectives, they often have to readapt it during the project development process.
the development process, developers and users can exchange knowledge by jointly exploring the full potential of a new system (Barcelli et al., 2008; von Hippel, 2001).

Developers are defined as project team members who are directly registered on the project profile as developers who directly contribute to the project development. They are involved in different project activities, such as programming work, reviewing the source code, and detecting and fixing bugs (Ghanpanchi, 2013). Developers have a central role, as they are the primary source of an emerging OSS project. However, previous studies in the OSS literature commonly recognize that users play a critical role in the evolution of open source products (e.g., Von Hippel & Von Krogh, 2003; Singh et al., 2011), as they strongly contribute to the modification and considerable improvement of such products. Numerous studies have developed measures based on user contributions to capture OSS project popularity. For example, user satisfaction, which is measured on the basis of users’ interest in a project according to user ratings and via user surveys or user opinions on mailing lists, is a commonly used measure of system performance (Crowston et al., 2003, 2006). Other measures focus on the development process, such as the level of activity of users’ contributions (Crowston et al., 2006). However, in the context of OSS, user involvement is hugely complex because user involvement may be used only as a buzzword or weapon for achieving solely managerial ends (Iivari 2009a). In the context of OSS, a user’s role can be divided into two categories: technical readers and non-technical readers, which can act as consultative and participative users, respectively, or only as consultative users (Iivari 2009a). Users are consultative only when their role consists of commenting on predefined design solutions, whereas participative users actively participate in the design process and are also decision makers (Damodaran, 1996; Iivari, 2009a). More recently, Sutanto et al. (2014) similarly differentiated between active and passive users, explaining that active users provide answers to queries posted in the community discussion forum, whereas passive users are community members who post only knowledge-acquiring messages. The authors demonstrated that this effective online user support community is also necessary to foster OSS use and popularity by contributing to OSS development and to influence the user community either positively or negatively (Sutanto et al., 2014). In particular, users...
interest and involvement as well as user communication about an OSS project have been proven to have a positive impact on OSS popularity (Ghapanchi, 2013). Thus, it is clearly recognized that both developers and users have identified their roles in the development of an OSS project and have considerable impact on OSS project performance. In other words, the interaction between users and developers can reinforce the mutual adaptation that occurs during project development to meet users’ expectations. To better understand the role of both users and developers, we argue that we must focus intensely on the nature of the interactions between users and developers and their impact on OSS project popularity. Based on prior research studies, we argue that the user-developer interaction is an important factor that affects OSS popularity. To capture the level of the user-developer interactions, we focus on two elements. First, the OSS literature generally uses the exchange of information, or information flow, among these participants. Information flow represents the level of exchanges between co-workers and leverages the knowledge of others within the community (Sharma et al., 2002). Other studies have recognized that the flow of information among OSS members (developers and users), which is indicative of the level of interaction within the community, plays a critical role in the way in which problems are solved (Von Hippel & Von Krogh, 2003), bugs are fixed, patterns are identified and collaboration is carried out (Xu et al., 2007). Using social network analysis, academics have demonstrated that such communication among members has a positive impact on OSS success (Miralles et al., 2006; Wu et al., 2007; Stuermer et al., 2009; Iivari, 2009a Benkeltoum, 2013). In the context of our study, we argue that information flow is a valuable indicator of the level of interaction between users and developers with respect to its impact on OSS popularity. Thus, we posit the following hypothesis:

**H1a: Information flow among users and developers during OSS activities will positively influence OSS popularity.**

The OSS literature has examined “bug-related activities” to understand the role of OSS communities in OSS success. More specifically, scholars have focused on whether OSS users and developers test new releases, submit bug reports, request features, and help others install, configure, and use software (Zhang et al., 2013, Choi et al., 2015). Thus, examining activities related to fulfilling bug reports and feature requests might yield useful data regarding a project’s status (Crowston et al., 2003). The number of bugs solved has been shown to influence the quality of software (Crowston et al., 2003; Sohn & Mok, 2008). For instance, academics have measured OSS project success by using bug-related activities such as the number of bugs fixed (Grewal et al., 2006), the time required to fix bugs (the ratio between fixed bugs and total bugs) (Crowston et al., 2006), and the number of bug reports (Lakhani & Von Hippel, 2003). This process of contributing to the code and fixing bugs has been demonstrated to be carried out in an iterative manner (Midha et al., 2010). Indeed, the effectiveness of the support community captures the interaction between users and developers, as users’ activities are often represented by bug-reporting activities and a greater number of people working on the code allows developers to identify and fix bugs more quickly (Stewart & Gosain, 2006). We argue that bug-related activities constitute a good indicator of the quality of interaction among users and developers, which influences OSS popularity. Thus, we posit the following hypothesis:

**H1b: The time required to fix bugs during OSS development will positively influence OSS popularity.**
**H1c: The total number of fixed bugs during OSS development will positively influence OSS popularity.**

### 2.3.2. Market potential

The notion of market potential has been used as an indicator of general project interest (e.g., Grewal et al., 2006). OSS market potential can be explained by using network embeddedness theory (Sutanto et al., 2014), which suggests that potential users can be influenced by community-provided user support regarding product use (voluntary contributions within an OSS community). As a result, a more active user community often improves perceptions of a project because of the resulting positive word of mouth (Van den Bulte & Lilien, 2001), which generally increases the potential and viability of the project. The notion of market potential has often been associated with understanding OSS global diffusion among potential end-users to drive dynamic diffusion (Lakka et al., 2012).

Market potential, therefore, recognises the interest level of potential users for a particular project. Different factors can drive the market potential of OSS projects. For example, Lakka et al. (2012) analysed socio-economic factors or conditions and their relations with OSS market saturation. They found that high user interest is positively associated with institutional quality. Other social factors, however, such as pressure from end users to increase the social responsiveness of a programme (Miralles et al., 2006), have demonstrated that active user-developer interaction may be able to mobilize other volunteers to respond to some of the user queries, which may increase a project’s market potential and, consequently, its popularity (Sutanto et al., 2014).

Project success varies with the number of page views (how many pages of each OSS project have been visited) and number of visits (how many visitors have looked at the OSS project) as indicators of the market potential of a project, as the numbers of page views and visits directly indicate the general interest level in the project and its market potential (Grewal et al., 2006). Hence, when the number of queries increases, the number of page views and visits on the OSS project website increase (Singh et al., 2011; Sutanto et al., 2014) and the market potential of the OSS project rises, which positively affects its popularity. In other words, when there is positive word of mouth within the network of users, the number of page views and visits will increase (Van den Bulte & Lilien, 2001). In contrast, negative word of mouth dissuades users from visiting the project website, thus lowering the number of page views and visits (Grewal et al., 2006). Thus, the number of page views and the number of visits are good indicators of market potential. This assumption allows us to propose the following hypothesis:

**H2a: Total visits will have a positive impact on OSS popularity.**

**H2b: Total page views will have a positive impact on OSS popularity.**

### 2.3.3. Development stage

Recently, Sutanto et al. (2014) found that the development stage of an OSS project affects the number of downloads, which affects OSS popularity. In other words, various studies have shown that the number of downloads increases when OSS projects reach the production stage (i.e., the mature stage). Indeed, when a project is mature and advances to a later stage of development, the project is stable, and it can accumulate a greater number of important outcomes (user-developer interaction) that influence its popularity (Stewart & Gosain, 2006). As explained earlier, developers must always improve their product so that they meet...
users’ expectations, account for market evolution and resolve bugs. Wynn (2004) found that the fit between the satisfaction and involvement of both developers and users during the life cycle stages of an OSS project constitutes an indicator of project success. In addition, Stewart and Gosain (2006) explain that both input and output measures of effectiveness are functions of development stage due to the unpredictable utility of success in an earlier development stage. As we want to analyse the development processes that lead to popularity (including input, process and output), we need to account for the potential evolution through different stages of development of lesser-known OSS projects and the impact of these changes on popularity. We thus propose the following hypothesis:

H3: The development stage and maturity of an OSS project will have a positive impact on OSS popularity.

2.3.4. d. Number of developers

Stewart and Gosain (2006) examined the effect of the number of developers on OSS project success. We thus included the number of developers in our hypothesis testing because an OSS project with more developers may be downloaded more often (Sutanto et al., 2014). Controlling for the number of developer on each project allowed us to account for the level of human capital actively involved in a project (Singh et al., 2011).

3. DATA AND RESEARCH METHODS

We collected data from 657 OSS projects available on the SourceForge hosting platform. We decided to use one specific category of OSS projects, namely, enterprise applications that were exclusively hosted by the SourceForge.net website. The subcategory of enterprise applications concerned includes CRM (22%), ERP (33%), business intelligence (17%), data warehousing (15%) and workflow (13%) projects. The OSS projects we selected were considered lesser-known projects because, for each category, we decided to retain only projects in which the number of downloads was under 10,000. For example, we excluded projects such as open bravo and Sugar CRM, which have each been downloaded more than 10,000 times.

Development stage: We used the seven development stages provided on Sourceforge.net at the time of inquiry: 1) Planning; 2) Pre-Alphas; 3) Alphas; 4) Betas; 5) Production/Stable; 6) Mature; and 7) Inactive.

Developer-user interaction: The bug-related activities and information flows were collected from the forum of the selected OSS projects. Within these forums, we were able to collect data on the time required to fix bugs (the difference between when the bug was posted on the forum and when it was closed), on total bugs (the total number of bugs created on the forum) and information flow (the number of exchanges, i.e., messages, posts, and replies, on the forum).

Market potential: To evaluate the notion of market potential for the selected OSS projects, we measured visitors’ input in terms of number of page views (Total page views for each OSS project) and number of visits (how many people visited the OSS project page).

Popularity: The notion of popularity is typically defined as the general level of user interest for a project (Sutanto et al., 2014), indicating that users download or do not download the OSS. Following Crowston et al. (2006), popularity is thus measured by the number of downloads.

All variables are presented in Table 3.
Hereafter you will find a table representing the descriptive Statistics of our sample. To analyze the data collected, we have first performed a correlation to evaluate the relationships between the studied variables (Table 5). Then we assessed the collinearity through a multivariate regression of these variables (Table 6).

Once done, we have applied a specific PLS approach: the PLS regression as a methodology to appraise the conceptual model. The PLS approach (PLS-SEM) allows us to analyse a set of J blocks of variables. Each block can be represented by a latent variable. The PLS approach (PLS path modelling) developed by Herman Wold (1975, 1983) as an alternative to Lisrel, estimates structural equation models by linking several blocks of variables ($J > 2$) between them. Whereas, The PLS regression links one block of dependent variables $Y$ to one block independent variables $X$ ($J = 2$). We have decided to use the PLS regression to overcome certain constraints of classical linear regression (Wold et al., 1983) and to allow us to assess simultaneous the modeling of both structural and measurement models (Chin et al., 2003). In addition, PLS regression method presents many advantages and gives good results precisely when there is collinearity between variables (Tenenhaus, 1998). To avoid loss of information caused by other statistical procedures which proceed by the iterative elimination of the collinear variables. The PLS regression, makes it possible to retain the collinear variables of

<table>
<thead>
<tr>
<th>OSS Project Characteristics</th>
<th>Variables</th>
<th>Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Popularity</td>
<td>Number of downloads</td>
<td></td>
</tr>
<tr>
<td>Market potential</td>
<td>Total visits</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total page views</td>
<td></td>
</tr>
<tr>
<td>Developer-user interaction</td>
<td>Information flow</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total bugs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Time required to fix bugs</td>
<td></td>
</tr>
<tr>
<td>Development stage</td>
<td>Development stage</td>
<td></td>
</tr>
<tr>
<td>Number of developers</td>
<td>Number of developers</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3: Variables of OSS project popularity**

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of developers</td>
<td>1.68</td>
<td>[1-28]</td>
</tr>
<tr>
<td>Informations flow</td>
<td>5.52</td>
<td>[23-657]</td>
</tr>
<tr>
<td>Total bugs</td>
<td>0.79</td>
<td>[6-156]</td>
</tr>
<tr>
<td>Time required to fix bugs</td>
<td>869.89</td>
<td>[15-91109]</td>
</tr>
<tr>
<td>Total visits</td>
<td>6840.99</td>
<td>[254-245325]</td>
</tr>
<tr>
<td>Total page views</td>
<td>7092.84</td>
<td>[354-675542]</td>
</tr>
<tr>
<td>N</td>
<td>657</td>
<td>657</td>
</tr>
</tbody>
</table>

**Table 4. Descriptive Statistics**
the observed variables $X_i$, using for this the least partial squares minimizing the residual variance. The results are therefore closer to the observed reality.

To perform the PLS regression, we used SIMCA-P+ 12.0 and we have followed four steps to analyze the data collected: Step 1: we established both the block of dependent variables $Y$, and the block of independent variables $X_i$; Step 2, we performed the PLS regression to control the fit of the model per component (four model were assessed); Step 3: We assessed the model quality with indicators $R^2_Y$ (cum) and the $Q^2$ (cum) that are $\geq 0.5$; Step 4: we identified the variables of the PLS model that are significant using the VIP criteria.

4. RESULTS

We begin by analysing the correlation matrix that shows the potential relationships between the variables (Table 5). There is a significant and positive correlation between the total visits and all the other factors, where the coefficients range from 0.08 for total bugs to 0.466 for information flow. Concerning total page views, the correlations are positive and significant for three of the five factors but not significant for total bugs and time required to fix bugs. For information flow, we found that all the coefficients are positive and significant. Moreover, the variable total bugs is positively and significantly correlated with time required to fix bugs and development stage but not correlated with number of developers. However, we found a very strong link between time required to fix bugs and total bugs (0.983). Thus, we can conclude that collinearity is present between these variables. Regarding time required to fix bugs, the coefficient is significant and positive at 0.1 for development stage and not significant for number of developers. Finally, the number of developers is significantly and positively correlated with development stage (0.146).

<table>
<thead>
<tr>
<th></th>
<th>Total visits</th>
<th>Total Page Views</th>
<th>Information flow</th>
<th>Total Bugs</th>
<th>Time required to fix bugs</th>
<th>Number of Developers</th>
<th>Development stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total visits</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Page views</td>
<td>0.317**</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information flow</td>
<td>0.466**</td>
<td>0.196**</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Bugs</td>
<td>0.08*</td>
<td>0.008</td>
<td>0.078*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time required to fix bugs</td>
<td>0.09*</td>
<td>0.016</td>
<td>0.094*</td>
<td>0.983**</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of Developers</td>
<td>0.308**</td>
<td>0.169**</td>
<td>0.145**</td>
<td>0.050</td>
<td>0.046</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Development stage</td>
<td>0.277**</td>
<td>0.135**</td>
<td>0.127**</td>
<td>0.099*</td>
<td>0.100*</td>
<td>0.146**</td>
<td>1</td>
</tr>
</tbody>
</table>

**. Correlation is significant at the 0.01 level.
*. Correlation is significant at the 0.05 level.

Table 5. Correlation matrix
As mentioned above when we examined the correlation matrix, some variables were strongly correlated. To verify the presence of collinearity between these variables, we performed a multivariate regression to assess the collinearity statistics (tolerance and VIF). Tolerance measures the part of the variance in parameter estimates that result from the correlation between variables. A tolerance value in excess of 0.10 generally confirms the absence of collinearity (Kleinbaum et al., 1998). The VIF (Variance Inflation Factor) is defined as the reciprocal of tolerance. The acceptable levels of VIF have been recommended with a value of ten being the maximum level of VIF (Hair et al., 1995).

However, a recommended maximum VIF value of five (Rogerson, 2001) and even four (Pan & Jackson, 2008) can be found in the literature. A VIF value in excess of ten generally confirms the existence of collinearity. The results confirm the existence of collinearity between some variables (Table 5). The results regarding the collinearity statistics in Table 6 show that collinearity is present for total bugs (Tolerance=0.03 &VIF=29.12) and the time required to fix bugs (Tolerance=0.03 &VIF=29.18). The tolerance for these variables is approximately zero (under 0.5), and the VIF is higher than two, which confirms collinearity. However, to include all the observed variables and to avoid the exclusion of total bugs and the time required to fix bugs, we incorporated them together in an efficient econometric model using a partial least squares (PLS) regression.

The expression of our model is as follows:

\[
\log (\text{Number of downloads} + 1) = \beta_0 + \beta_1 \log (\text{Total visits} + 1) + \beta_2 \log (\text{Total page views} + 1) + \beta_3 \log (\text{Information flow} + 1) + \beta_4 \log (\text{Total bugs} + 1) + \beta_5 \log (\text{Time required to fix bugs} + 1) + \beta_6 \log (\text{Development stage} + 1) + \beta_7 \log (\text{Number of Developers} + 1) + \epsilon.
\]

Hereafter, we verify the validity of the PLS regression over all the independent variables Xi (total visits, total page views, information flow, total bugs, time required to fix bugs, development stage, number of developers) and the dependent variable Y (OSS project popularity – i.e., number of downloads). The objective of our analysis is to test the existence of a causal link between

<table>
<thead>
<tr>
<th>OSS project popularity (Download numbers)</th>
<th>Collinearity statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Tolerance</td>
</tr>
<tr>
<td>Total visits</td>
<td>0.65</td>
</tr>
<tr>
<td>Total page views</td>
<td>0.89</td>
</tr>
<tr>
<td>Information flow</td>
<td>0.77</td>
</tr>
<tr>
<td>Total bugs</td>
<td>0.03</td>
</tr>
<tr>
<td>Time required to fix bugs</td>
<td>0.03</td>
</tr>
<tr>
<td>Development stage</td>
<td>0.91</td>
</tr>
<tr>
<td>Number of developers</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Table 6. Collinearity statistics
OSS popularity, market potential, developer-user interaction, development stage and number of developers (a control variable) to assess OSS performance.

The quality of the model represents the trade-off between the PLS regression and the collected data. The quality of the model is confirmed and is considered satisfactory when the model with the original variables have $R^2(Y_{(\text{cum})})$ and $Q^2(cum) \geq 0.5$ (Tenenhaus, 1998). In our case, Model 4, which corresponds to our complete research model, explains 61% of the OSS project popularity variance by the independent variables $X_i$ ($R^2(Y_{(\text{cum})}) = 0.61$). The PLS regression method also allowed us to obtain a model with two axes that can predict 60% of the popularity variance ($Q^2(cum) = 0.60$). Thus, we can interpret that the phenomenon explaining $Y$ is well-framed by the $b$ axes (named component) $t_1, t_2, \ldots, t_b$ because $Q^2(cum) \geq 0.5$. The application of this rule indicates that our model with the original variables is satisfactory.

We determine the significance of the PLS components on the axes by estimating the weight of each variable, which we named $W_i^*c(1)$ for axis 1 and $W_i^*c(2)$ for axis 2. The proportion of the variance of the independent variables $X_i$ and dependent variable $Y$ are explained by the first axis $t_1$, which accounts for 55.2% of the variables $X_i$ and 59.7% of the variable $Y$. The second axis $t_2$, in turn, accounts for 12.8% and 1.74% of the variance of independent variables $X_i$ and dependent variable $Y$, respectively. Hence, the first axis better explains the variance in the model. Even if the axis 2 accounts for 12.8% and 1.74% of the variance, it's important to keep the second axe to represent the OSS projects in the factorial plan. If we do not take into consideration the second axe, we are going to miss some explaining and reduce the quality of the collected data. As shown in the Hotelling ellipse, the majority of OSS projects are clearly represented by $t_1$ and $t_2$ components of our PLS regression. So, we cannot just dismiss the second axe even if it accounts low in term of variance explained. Keeping the second axe allows us to have a better understanding of the PLS components on the axes by estimating the weight of each variable, which we named $W_i^*c(1)$ for axis 1 and $W_i^*c(2)$ for axis 2. The results are presented in Table 7.

<table>
<thead>
<tr>
<th>Variables</th>
<th>$W_i^*c(1)$</th>
<th>$W_i^*c(2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total visits</td>
<td>0.474</td>
<td>0.020</td>
</tr>
<tr>
<td>Total page views</td>
<td>0.456</td>
<td>-0.063</td>
</tr>
<tr>
<td>Information flow</td>
<td>0.345</td>
<td>0.692</td>
</tr>
<tr>
<td>Total bugs</td>
<td>0.228</td>
<td>-0.198</td>
</tr>
<tr>
<td>Time required to fix bugs</td>
<td>0.193</td>
<td>-0.447</td>
</tr>
<tr>
<td>Development stage</td>
<td>0.486</td>
<td>0.285</td>
</tr>
<tr>
<td>Number of developers</td>
<td>0.346</td>
<td>-0.452</td>
</tr>
<tr>
<td>Popularity ($Y$)</td>
<td>0.394</td>
<td>0.161</td>
</tr>
</tbody>
</table>

Table 7. Weight of variables
The importance of the axes can be determined by estimating the weights of the independent variables based on the construction of the axes from running the PLS regression. We note that market potential, measured by “total visits” and “total page views”, and development stage highly contribute to the first axis. Of course, the other variables also contribute to the first axis. Further, the second axis is mainly influenced by developer-user interaction, as measured by “information flow” and “time required to fix bugs”. However, we observe that these variables have opposite effects. The variable number of developers also contributes to the construction of axis 2. We note the existence of a relationship between “total visits”, “total page views”, “information flow”, “development stage” and “OSS project popularity”. Thus, the following variables have a significant causal link with OSS project popularity: “market potential”, “developer-user interaction” and “information flow”.

As presented in Table 8, we have performed an analysis showing the value of the model quality for each component for the full model (Model 4). We see that for $R^2(X)\text{ (cum)}$, $R^2(Y)\text{ (cum)}$ and $Q^2(\text{cum})$ are > 0.5 and that the second axe explain part of the model.

As the results presented in Table 9 show, the VIP (very important variables) indicate that all the variables contribute to the phenomenon studied. In particular, the results reveal that three factors – market potential, developer-user interaction (only for information flow) and development stage – have a positive and significant influence on OSS project popularity. The results specifically show that development stage, information flow (capturing developer-user interaction), total visits and total page views (both of which capturing market potential) explain 24%, 25%, 19% and 17%, respectively, of OSS project popularity.

We have assessed our model by using a PLS regression model applied to the 657 projects from which we collected data (i.e., enterprise applications). The analysis of OSS project popularity is based on the number of developers, market potential, developer-user interaction, and development stage. The analysis shows that the number of developers has a moderate influence on the OSS projects popularity significance with a VIP = 0.92 (between 0.8 and 1) (Olah et al. 2004: 442). (cf. Table 7). We consider this effect to be negligible as the coefficient is (0.064). Moreover, the effect of market potential (total visits and total page views) is highly influential in term of significance.

(VIP = 1.32 and VIP = 1.26, respectively) as their VIP is over 1 (Olab et al., 2004: 442; Tenenhaus 1998: 183). Their coefficient 0.19 and 0.17 respectively show that these

<table>
<thead>
<tr>
<th>Components</th>
<th>Model 4</th>
<th>Nbr Obs: 657</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$R^2X$</td>
<td>$R^2X(\text{cum})$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0.552</td>
<td>0.552</td>
</tr>
<tr>
<td>2</td>
<td>0.128</td>
<td>0.680</td>
</tr>
</tbody>
</table>

Table 8. Value of the model quality
Concerning developer-user interaction, the coefficient is significant at 0.25, indicating that developer-user interaction positively influences OSS project popularity, supporting Hypothesis 2.

By contrast, H1b and H1c are not supported, as the coefficients for time required to fix bugs and total bugs are not significant. Finally, H3 is also supported, as the coefficient for development stage is significant at 0.24, indicating that this variable positively and moderately influences OSS project popularity and partially supporting Hypothesis 1 (i.e., H1a is supported).

Table 9. Weight of the independent variables of the PLS regression

*VIP = Importance of the independent variables in explaining OSS project popularity.
*Coef = Weight of the regression coefficient for the dependent variable.
*Sign 0.05 (5%).
strongly influences OSS popularity. Thus, we can conclude that we have found support for most of our hypotheses related to our research questions, as our results show that OSS project popularity is explained by market potential, developer-user interaction, and development stage.

To further analyse the impact of the development stage on OSS popularity and complete our understanding of our research model, we divided the sample of OSS projects based on their development stage, with values between 1 and 3 indicating younger projects and those between 4 and 7 indicating more advanced and mature projects. We then re-applied the PLS regression to a new model, defined as Model 5, that included the same variables as Model 4 but used two different samples (younger projects and more advanced projects in terms of development). According to the results of Model 5, the $R^2_Y$ was 0.36 for younger projects and 0.77 for more advanced projects, where the difference was significant at $p < 0.05$. Thus, our model better explains OSS popularity for more advanced projects than for younger projects, which may be predicted from the finding that younger projects have more bugs and are less stable than advanced and mature projects. It seems to be incorrect. Similarly, Number of developers and Information flow are moderately influential because VIP (respectively 0.92 and 0.95) are between 0.8 and 1 (Olah et al., 2004: 442). Finally Total bugs and Time required to fix bugs are less influential because their VIP are $< 0.8$ (Lee et al. 2011: 210; Pérez-Enciso et Tenenhaus 2003: 586).

5. DISCUSSION

The present research addresses the following research questions: Does the combined effect of user-developer interaction and market potential lead to greater OSS popularity? Moreover, how does the development stage influence this combined effect on OSS popularity in lesser-known projects? To conduct this study, we investigated the combined effect of these three factors on OSS popularity. Our developed model was applied to data from a sample of 657 open source projects hosted by Sourceforge.net with a focus on the development of
Based on our data analysis, we can conclude that OSS popularity is positively influenced in part by user-developer interaction, by market potential, and by the development stage during OSS development. Previous studies demonstrate that the boundaryless freedom of OSS offers opportunities for better communication channels (Stewart & Gosain, 2006) and multi-level interactions between the system and surrounding environment (Aksulu & Wade, 2010). In addition, Ghapanchi et al. (2014) highlight that extensive communication between users and developers, especially through active forums, leads to increased project effectiveness. Consistent with the current literature, our research complements these findings. Our results show that the level of information exchange via forums, between users and developers, can positively impacts OSS popularity. In other words, extensive information exchange, from multiple interactions between users and developers, is optimal for knowledge sharing (Osterloh & Rota, 2007) and is associated with greater OSS popularity.

Nevertheless, the lack of significance for the effect of bug-related activities on OSS popularity is interesting and needs to be highlighted. Numerous studies have proposed the time required to fix bugs to be a measure of software quality (e.g., Kim et al., 2005), also demonstrating that the total number of bugs and the time required to fix them significantly influence OSS success (e.g., Grewal et al., 2006; Ghapanchi et al., 2014). Our study showed that while information flow has a moderative effect on OSS popularity, bug-related activities do have a less significant effect on OSS popularity. As OSS popularity is used to measure OSS success for lesser-known projects (Crowston et al., 2012), interestingly, our results reveal that in such a context, and in contrast to the existing literature, bug-related activities do not have an impact on OSS popularity and do not constitute a valuable measure for project popularity. This controversial result could be explained by the fact that we focused on lesser-known projects, where users expect a high level of communication rather than high quality on technical aspects. In other words, the level of attention and interest is more important than the level of activities from the developers of the project (Stewart et al., 2005).

Further, to strengthen the measure of OSS popularity, market potential must be considered. Previous research has shown that OSS project success increases as the number of visits increases (Grewal et al., 2006) because positive word of mouth increases the number of user visits to webpages. Consistent with this finding, our results showed that the number of total visits increases with the number of total page views and significantly affects the number of downloads.

Moreover, another interesting finding is that the effects of user-developer interaction and market potential on OSS popularity change as projects move through different development stages. This result corroborates previous research (e.g., Stewart & Gosain, 2006) showing that the development stage plays an important role in determining project popularity. Specifically, this result is consistent with the fact that the more the project evolves the more the community develops, ensuring a high level of communication based on the roles played by the co-members (Peng et al., 2013). Network robustness is positively related to later development stages and positively impacts project interest in the user-developer community (Grewal et al., 2006).

Our findings have some interesting implications for both theory and practice.
Implications for research

From a research perspective, academics have commonly recognized that the challenges involved in exploiting communal resources due to the particular OSS context lead many OSS firms to struggle for survival (Dahlander & Magnusson, 2005). Accordingly, numerous studies in the IT literature have aimed to explain OSS success by examining several variables separately. However, studies explaining OSS popularity focused on lesser-known projects remain limited, even if this topic is relevant. Our research highlights the combined effects of factors that influence OSS popularity in OSS projects.

First, the empirical results generally support the theoretical reasoning based on a social network perspective within the broader context of OSS user group participation rather than OSS development alone (e.g., Hars & Ou, 2002; Lakhani & Wolf, 2005; Bagozzi & Dholakia, 2006). Specifically, the level of information flow was shown to have a significant impact on OSS popularity, supporting our arguments that such interaction among user and developers increases users’ interest in a project and thus the number of downloads. Moreover, the unexpected result of a less significant impact for bug-related activities is relevant within our reasoning. Indeed, the level of communication inside virtual communities is a more valuable measure for OSS popularity than the numbers of developers involved in the community in the specific case of lesser-known project. Our “counterintuitive” finding thus highlights new factors that should be used to measure OSS popularity. This result is consistent with the fact that for lesser-known project, popularity, measured with the number of downloads, represents “how much interest and attention the project generates among current and potential users” (Stewart et al., 2005). In other words, the focus is not on technical aspects but rather on how to attract users. In this case, the level of communication and interaction appears to be more important to increase interest from potential users.

Second, another contribution of our study is that we clarify the role of market potential in influencing OSS project popularity. As explained above, positive word of mouth seems to positively influence the success of a project, as it provides a positive signal for potential users. Specifically, the number of page views and number of visits are also valuable indicators for users that influence their interest to lesser-known projects, which directly affects the popularity of such projects (Gallego et al., 2008).

Third, the OSS literature contains numerous works that have used the development stage (or project age) as a control variable because projects in earlier stages may be less certain to provide utility and may thus reduce motivations for input (e.g., Stewart & Gosain, 2006). The present research thus tested the direct impact of the development stage on OSS popularity. The obtained positive relation between OSS popularity and the development stage empirically indicates that the development stage of OSS projects is an important factor in measuring project popularity. This finding provides new theoretical insights regarding the factors that affect OSS popularity by highlighting a new factor that directly affects OSS popularity, especially for lesser-known project.

Implications for practice

Furthermore, our findings have some implications for organizations that use and adopt OSS development practices. The number of firms using OSS has increased in the past several years, although these projects do not always succeed. More important, firms are working on developing their own software, which often...
leads to the adoption of lesser-known projects for users. Research identifying the factors that influence OSS popularity can thus provide a preliminary understanding for organizations that want to better understand OSS development (Stewart & Gosain, 2006). In particular, our results indicate that firms should pay attention to the development stage of an OSS project before deciding to internally implement a new OSS project. Indeed, the results indicate that the maturity of projects influences the number of downloads, indicating that organizations should take into account the project advancement and maturity to limit the risk of failure while implementing an OSS project. Past research explains that the number of downloads is a relevant indicator of OSS project success, as this metric indicates the traffic on the project website (e.g., Stewart et al., 2005). However, in our study, because we focused on lesser-known projects, we used the number of downloads as an output. This indicator represents a highly relevant method of measuring the popularity of a project, which thus allows the evaluation of potential success. Because potential success depends on potential users, firms might benefit from better understanding of why users should be interested in using one project over another depending on the stage of development. Moreover, this approach would allow for adaptation and the readjustment of key elements during project evolution. Beyond providing guidance on the aspects that influence OSS popularity, the results show that firms should focus on multiple factors to avoid failure. Substantial attention has been devoted to increasing users’ and developers’ motivation to participate in a project (e.g., Meissonier et al., 2010) or attracting users and developers (e.g., Krishnamurthy, 2002). However, our study suggests that beyond the number of users or developers, the activities of these actors – that is, the level of information flow among users and developers – is directly related to OSS popularity. Firms should therefore be aware that simply attracting developers may not ensure project success (Stewart & Gosain, 2006), and managers should focus on developing quality interactions among members to foster OSS popularity. In addition, the number of pages viewed represents an important concern for firms, as it signals the general level of interest for a project. This measure of market potential for an OSS project should thus represent a valuable indicator of OSS popularity that firms may consider during OSS development or decision-making.

6. CONCLUSION

This research investigated the factors that affect OSS popularity in the context of lesser-known projects. The results revealed that user-developer interaction and market potential have a positive impact on OSS popularity. However, in contrast to previous OSS studies, information flow among users and developers has a stronger influence on OSS popularity than bug-related activities. The findings further highlight that our model is more significant with respect to advanced projects, demonstrating the importance of project development in explaining OSS popularity. Nevertheless, our research has some limitations that provide avenues for future research.

First, a main limitation of our research relates the variables we used in our model. Specifically, we investigated the combined effect of user-developer interaction, market potential and development stage on OSS popularity, and we controlled for the number of developers; however, we did not control for community size and project category. This limitation should be overcome in future research, as previous studies...
have suggested that these factors have a significance influence on OSS popularity (Sutanto et al., 2014).

Second, our model is not appropriate for determining the influence of individual factors on OSS popularity. Numerous studies have relied on individual factors that lead to the emergence of projects leaders within an OSS community and have linked these skills to OSS success (e.g., Lerner & Tirole, 2001; Giuri et al., 2008). It would be useful for future research to apply and transpose these individual factors to our model to increase our understanding of OSS popularity.

Third, this study focused on lesser-known projects to evaluate the factors that affect OSS popularity (Crowston et al., 2012), whereas previous research in the OSS literature used the notion of success to evaluate the quality of lesser-known projects. However, some recent research has assumed that the interrelationships among different success indicators of OSS projects are not well understood in the literature (Ghapanchi, 2015). Accordingly, it would be useful to replicate the analysis in this study on more familiar projects to provide a better understanding of the general measures of OSS quality.

Despite these limitations, our research provides some interesting implications for both theory and practice. In particular, we reveal some unexpected patterns that add to our understanding of why certain OSS projects are popular by employing a novel perspective that combines factors that affect OSS popularity. Our results should thus encourage researchers studying open source systems to adopt a perspective that more integrates success and popularity factors to increase knowledge on OSS as a part of innovation strategy (Teigland et al., 2014).

REFERENCES


Stewart, K.J., Ammeter, A.P., Maruping, M.L. (2005), "A Preliminary Analysis of the Influences of Licensing and Organizational Sponsorship on Success in Open Source Projects", in Proceedings of the 38th Hawaii International Conference on System Sciences, Big Island, HI.


of the International on Information Systems, Montreal, Canada.


