

RESEARCH

Open Access



Supporting governance of mobile application developers from mining and analyzing technical questions in stack overflow

Awdren Fontão^{1*} , Bruno Ábia¹, Igor Wiese², Bernardo Estácio³, Marcelo Quinta⁴, Rodrigo Pereira dos Santos⁵ and Arilo Claudio Dias-Neto¹

* Correspondence: awdren@icomp.ufam.edu.br

¹Institute of Computing, Federal University of Amazonas, Manaus, AM, Brazil

Full list of author information is available at the end of the article

Abstract

There is a need to improve the direct communication between large organizations that maintain mobile platforms (e.g. Apple, Google, and Microsoft) and third-party developers to solve technical questions that emerge during the project and development of developers' contributions in a Mobile Software Ecosystem (MSECO). In this context, those organizations may not know how to define and evolve strategies to govern their developers towards achieving their organizational goals. Such organizations use an infrastructure to support developers, for example, questions and answers (Q&A) portals such as Stack Overflow. Interactions among developers in these portals feed a Q&A repository that can serve as a mechanism to understand and define strategies to support developers. In this paper, we mined 1,568,377 technical questions from Stack Overflow related to Android, iOS, and Windows Phone platforms. Next, we performed comparisons among those MSECO regarding: (i) developers' activity intensity, (ii) hot-topics (using Latent Dirichlet allocation algorithm) from all and more commented/viewed questions, (iii) "What" and "How to" questions, (iv) hot-topics from more viewed unanswered questions, and (v) relationship among questions and official developer events. From the results, we identified four key insights: recruiting, educating, and monitoring strategies; barrier reduction; management of technology insertion; and fostering of relationships. The relevance of the four key insights to support developer governance was evaluated by practitioners through a survey. Finally, for each key insight we associated a total of 10 strategies to support developer governance activities. Such strategies were extracted from 65 studies identified through a systematic mapping of the literature.

Keywords: Software ecosystems, Mobile application development, Mining software repository, Stack overflow

1 CCS concepts

- Software and its engineering → Software creation and management → Collaboration in software development

2 Introduction

In Software Engineering, the relationship among mobile application developers and an organization responsible for a technological platform (keystone) which involves cooperation and competition has been investigated as a Software Ecosystem (SECO) (Bosch, 2009). In the mobile application (app) scenario, this context refers to a specific type of SECO, known as Mobile Software Ecosystem (MSECO) (Lin & Ye, 2009; Fontão et al., 2015). The developer is an essential actor to sustain MSECO contributions, such as apps and technical documentation (Fontão et al., 2016; Koch & Kerschbaum, 2014). Such contributions are usually stored in an official, internal MSECO repository, such as Android¹ and Apple² Developers repositories. Moreover, there are links between MSECO and external repositories, such as code (e.g. GitHub) (Casalnuovo et al., 2015) and questions and answers (Q&A) repositories. As an example of an external Q&A repository, Stack Overflow has a set of technical questions/answers that arise from the use of APIs, SDKs, and development tools (Ahmad et al., 2018). Such external repositories help to maintain the interaction among developers over a common platform, resulting in a set of contributions and influencing directly or indirectly the ecosystem as a whole (Santos & Werner, 2012). Therefore, Stack Overflow has archived communications among ecosystem developers and it can be used to investigate some MSECO aspects, for example, developer engagement and code snippets.

A self-sustained MSECO (in which the attraction, onboard and engagement of developers take place) helps users, supports organizational goals and depends on external repositories (Wareham et al., 2014). In this context, developer governance is a set of mechanisms to support win-win relationships between a thriving developers' community and an organization aiming to insure/monitor developers' economic and social welfare. The keystones maintain Developer Relations teams work closely with developers supporting them in their activities and contributions.³ Ineffective governance can result in a declining growth of the ecosystem (Wareham et al., 2014); for example, Windows Phone MSECO officially died in 2017 because developers never backed the platform.⁴ In this work, we analyze the MSECO Windows Phone as a way to extract indicatives of the possible "death" of the ecosystem.

Manikas (2016) argues that more in-depth (instead of in-width) studies are necessary in the MSECO context. Focusing on a specific subset or type of ecosystem in depth would arguably be a challenge that is easier to tackle in order to bring results that are more realistic rather than wide ecosystem studies focusing on a single aspect (e.g. architecture). Considering the existing literature reviews on SECO (Manikas, 2016; Barbosa & Alves, 2011; Manikas & Hansen, 2013a), MSECO (Fontão et al., 2015), and Q&A repositories (Farias et al., 2016), there is no indication of studies that investigate the use of a Q&A repository to understand an MSECO. However, there is an indication of using approaches of mining software repositories as a way to extract information about the socio-technical perspective of a SECO (Manikas, 2016). In this context, Stack Overflow is a community of five million registered developers with 3.9 billion visits. This reality puts us at the forefront of a research question: "*What can be understood from the three main MSECOs (i.e. Android, iOS, and Windows Phone) based on technical questions at Stack Overflow?*" It can help a keystone to get a good overview of the ecosystem providing effective measurements to analyze developer performance (Eckhardt et al., 2014) and to assist in developer engagement as an instrument for supporting developer governance.

In our study, we mined Stack Overflow to investigate the abovementioned research question. Our work was inspired by studies performed by Bajaj et al. (Bajaj & Mesbah, 2014) who analyzed technical questions asked by web developers in Stack Overflow, Barua et al. (2014) (focus: testing), and Rosen and Shihab (2016) (focus: mobile developers). The contributions of this paper are a set of four key insights and 10 developer governance strategies that can help keystones to understand developer engagement in an MSECO. The four key insights were evaluated regarding the relevance level to developer governance in MSECOs. Then, strategies were associated to key insights from 65 studies of existing technical literature on developer governance in order to indicate practical ways for the execution of key insights (Fontao et al., 2017).

This paper is organized as follows. Section 3 presents background. Related work is discussed in Section 4. In Section 5, we explain the research methodology used in this work. In Section 6, we describe the empirical study involving the mining Stack Overflow. Section 7 presents results' analysis as well as the survey with practitioners on the relevance of the key insights and on developer governance strategies. Finally, Section 8 concludes the paper and points out future work.

3 Background

This section covers the concepts of MSECOs, developer governance, mining questions and answers repositories and Stack Overflow.

3.1 Mobile software ecosystems

An MSECO comprises several elements surrounding a mobile app or simply app (Fontão et al., 2015). An important issue is the relationship among elements (e.g. developers, keystone, and users) that result in technical (e.g. apps, sample codes) and non-technical (e.g. user reviews of an app) contributions (German et al., 2013). An MSECO is a specific SECO for mobile applications that will be shipped on mobile devices. The activity of each developer in an MSECO is motivated by value creation for both the developer and the ecosystem. To support the developer within MSECO, organizations have a team of professionals working within an area called Developer Relations (DevRel)³. DevRel involves a group of software engineers who are outgoing and great at public speaking. It considers developer evangelism and advocacy and serves as an interface between developers and organization's platform product and technical teams. In this context, evangelism focuses on promotion and awareness. On the other hand, advocacy prioritizes gathering product feedback from developers.

From the knowledge perspective, an MSECO has a hybrid business structure, i.e. the ecosystem supports both proprietary and open source strategies to manage contributions (Manikas, 2016). A MSECO has a hybrid structure because it maintains a Mobile Application Store (a proprietary governance strategy to distribute extensions for MSECO users) and uses open-source-inspired strategies to attract, onboard, retain and recognize third-party developers. For instance, in the Windows Phone MSECO, the official support site indicates two sources working as technical forums: MSDN Forums (internal) and Stack Overflow (external). As pointed out by Souza et al. (2016), Q&A repositories as Stack Overflow support the interaction among developers from hybrid ecosystems (e.g. Android, iOS or Windows Phone, or simply WP, in this study).

From a technical dimension, a large amount of data is often readily available in those Q&A repositories, and the data is stable and not influenced by researchers (Shull et al., 2008). In this scenario, we can use methods from mining software repositories to conduct empirical studies in the ecosystem field (Farias et al., 2016). This method can help in defining and evolving strategies to govern developers in MSECO.

3.2 Developer governance

Alves et al. (2017) define software ecosystem governance mechanisms as managerial tools that aim to influence the ecosystem's health. Ecosystem's health refers to which extent a SECO is functioning well (Manikas & Hansen, 2013b). Specific measurements may be introduced to provide an overview of the state of the ecosystem while at the same time raise attention for actions and allow comparison of ecosystems. In this scenario, there are three main categories of governance mechanisms (Alves et al., 2017): *Value Creation* – to generate and distribute value; *Coordination* – to maintain consistency and integration of activities, relationships, and structures of an ecosystem; and *Organizational Openness and Control* – to capture tensions between open and closed models.

Governing MSECO requires a slight balance of control between platform provider and external developers (Song et al., 2018). Moreover, a well-chosen platform provides considerable competitive benefits, while a poorly-chosen one puts them at a disadvantage. In this scenario, Valença and Alves (2017) point out the need for understanding how platform governance affects MSECO innovation. In other words, understanding how the implementation of specific governance mechanisms affects the success of an ecosystem and its underlying enterprise platform is an exciting problem for researchers in the field.

Baars and Jansen (2012) state that ecosystem governance can help a company achieve its goals, make better use of available resources and can ultimately lead to increasing revenue and lower risks. However, since it is a relatively new field (Manikas, 2016; Mäenpää et al., 2017), many organizations do not know how to effectively manage their ecosystem, or even how to make their ecosystem ready for a governance strategy. Another point indicated by Axelsson & Skoglund (2016) is the difficulty in evaluating how data is used to govern platform ecosystems in practice (and how to generalize the findings). Therefore, research on ecosystem governance can help scholars and practitioners to address a topic that is highly relevant in practice (Schrieck et al., 2016).

However, since SECO is a relatively new field, many organizations may not know how to effectively manage their ecosystem, or how to get their SECO ready to begin with. Proper formalization for ecosystem governance is lacking and organizations concerning ecosystems have several challenges to overcome (Alves et al., 2017), for example, the attraction and engagement of developers. There is also a need for understanding developer governance.

3.3 Mining questions & answers repositories

Software repositories can be a valuable source of information since they contain (or may allow to extract) information about the technical and social perspectives of a software project, such as sources of developer communications (Genc-Nayebi &

Abran, 2016). Mining Software Repositories (MSR) area focuses on uncovering useful information about software by extracting and analyzing data from different software repositories (Ahmed, 2008). The unstructured data in software repositories have also pushed the Software Engineering research community to mine and analyze useful knowledge present in such repositories, i.e. different versioning systems (e.g. Git), archived communications (e.g. mailing lists), chat logs, online forums (e.g. Q&A repositories), mobile app stores (e.g. user reviews on Google Play) and online video-sharing websites (e.g. programming tutorials shared on YouTube) (Ahmad et al., 2018).

MSR approaches have been used for different goals, e.g. analyses of contribution and developer behavior. In this scenario, Q&A repositories are an important object of analysis. Q&A repositories are web, collaborative, social computing platforms that aims at supporting crowdsourcing knowledge by allowing users to post and answer questions. They not only provide a platform for experts to share their knowledge and be identified as community members, but also help newcomers to solve their problems effectively (Bhat, 2014). According to Shah et al. (2014), Stack Overflow would be an example. In this work, we use MSR techniques to mining unstructured data in such repository.

Developers of an MSECO who begin to interact within a Q&A repository such as Stack Overflow are generating knowledge for the developer community and can be actors involved in reducing barriers to promote engagement within the ecosystem. Once the core organization relies on a critical mass of third-party developers to meet user demands, value creation for products and MSECO sustainability, mining Q&A repositories can be useful as support for developer governance activities in an MSECO.

3.4 Stack overflow

Stack Overflow is a community-driven Q&A website used by developers who post and answer questions related to computer programming (Bhat, 2014) (approximately 14 M questions and 19 M answers³). This repository's questions and answers may receive users' votes (against/in favor of). Such votes become reputation points that allow developer to have some privileges, such as releasing restrictions on creating a publication and editing questions and answers from other users. Another privilege mechanism involves the assignment of badges, i.e. developer achievements while using Stack Overflow. Developers can get badges from several activities, for instance, a developer can receive a badge if he/she has asked a question that reached more than a thousand visits.

Zagalsky et al. (2016) identified that developers uses Stack Overflow for several reasons: (a) the ability to gain peer recognition; (b) its rich and user-friendly interface; (c) answers are straight to the point; (d) questions are usually answered faster than other forums; and (e) it is easy to search for previous questions and answers.

4 Related work

Bajaj & Mesbah (2016) presented a study of common challenges and misconceptions among web developers, by mining related questions over Stack Overflow. The authors used unsupervised learning (Latent Dirichlet Allocation, or LDA) to

categorize the mined questions and define a ranking algorithm to rank all the Stack Overflow questions based on their importance. The results indicated that the overall share of web development related discussions is increasing among developers, for example.

Barua et al. (2014) also used LDA to automatically discover the main topics from Stack Overflow dataset (July 2008 to September 2010) regarding developer discussions. Their analysis allowed them to make a number of interesting observations: developers' topics of interest range widely from jobs and version control systems to C# syntax; questions in some topics lead to discussions in other topics; and topics becoming more popular over time are web development (especially jQuery), apps (especially Android), Git, and MySQL.

Rosen and Shihab (2016) investigated what issues mobile developers ask about using data from Stack Overflow (updated in March 2013). The authors used LDA to summarize the mobile-related questions. Some findings were: app distribution, mobile APIs, data management, sensors, and context. They focused on identifying challenges faced by mobile developers. The authors motivate more research in this field as a way to improve mobile development processes.

Baars and Jansen (2012) proposed a framework, which consists of a questionnaire to diagnose the SECO governance from companies. With the framework, the company can gain strategic advantage over other companies by analyzing and improving governance in a structured way. The framework is composed of five parts: 1) Ecosystem clarity; 2) Clarity of the governance strategy; 3) Responsibility; 4) Measurement; and 5) Sharing knowledge. The proposal does not specifically analyze developer governance and also does not focus on MSECO.

Albert et al. (2013) proposed an approach to SECO governance that allows the organization to locate itself in the market and map its relationships with suppliers, distributors, products and technology through a tool called Brechó-SECOGov. The authors realized that the tool is applicable in the context of IT architecture to monitor the adoption of technologies by the organization. The focus is on the technical resources and perception of the consumer organization by the participants, but not on the analysis of a hybrid ecosystem such as an MSECO. In addition, the work does not focus on developers.

Sadi et al. (2015) proposed a generic approach based on Android and iOS ecosystems to identify types of developers and derive alternative solutions to design appropriate collaboration. For example, the authors have found that Android platform developers choose an open source platform to cultivate intrinsic motivations such as skill development and reputation enhancement. This study focuses on the objectives and decision criteria of the developers, but does not provide specific guidance on how these activities can be performed. To demonstrate the feasibility of the proposed recommendations, experimentation in real case studies is required.

Foerderer et al. (2018) examined the limits of knowledge involved in the governance of a proprietary SECO. They analyze several resources including developer support portals, documentation, and workshops. The analysis indicates that these resources help in defining the scope to which the knowledge will be directed, allowing the scalability of knowledge within the ecosystem. The objective of the authors was to analyze the resources used at the borders of the ecosystem to manage knowledge.

There is no specific analysis of developer governance from the perspective of used resources. In addition, the focus is on ecosystems in a general way, not on MSECO.

Considering the existing literature reviews on SECO (Manikas, 2016; Barbosa & Alves, 2011), MSECO (Fontão et al., 2015), and Q&A repositories (Manikas & Hansen, 2013a; Farias et al., 2016), there is no indication of studies that investigate the use of a Q&A repository to understand an ecosystem (and specifically an MSECO). However, there is an indication of using MSR techniques as a way to extract information on the social-technical perspective of ecosystems. Therefore, our study contributes to evaluate another source of information to analyze SECO and its elements (e.g. developers, repositories, platforms, and keystone) and provides key insights evaluated by practitioners and strategies to support developer governance in MSECOs.

5 Research methodology

Developer governance aims to support the synergy between the developer’s expectations and the keystone’s goals. The research methodology (Fig. 1) adopted for this research involves the analysis of developers’ perceptions in Stack Overflow and the point of view of the keystone about strategies to govern developers. The first allows us to get the information as result of developer interactions that emerges from Stack Overflow to understand the developer’s perspectives during engagement in an MSECO. The second directs us to the understanding of how the organization perceives the relevance of strategies to govern developers. As such, the studies are:

- 1) **Mining Technical Questions:** in this stage of the research, the objective was to analyze the behavior of developers in a repository of questions and technical answers for the extraction of key insights for the governance of developers. The chosen repository was the Stack Overflow because it is the largest Q&A repository. Related key insights have gone through a process of peer review performed by

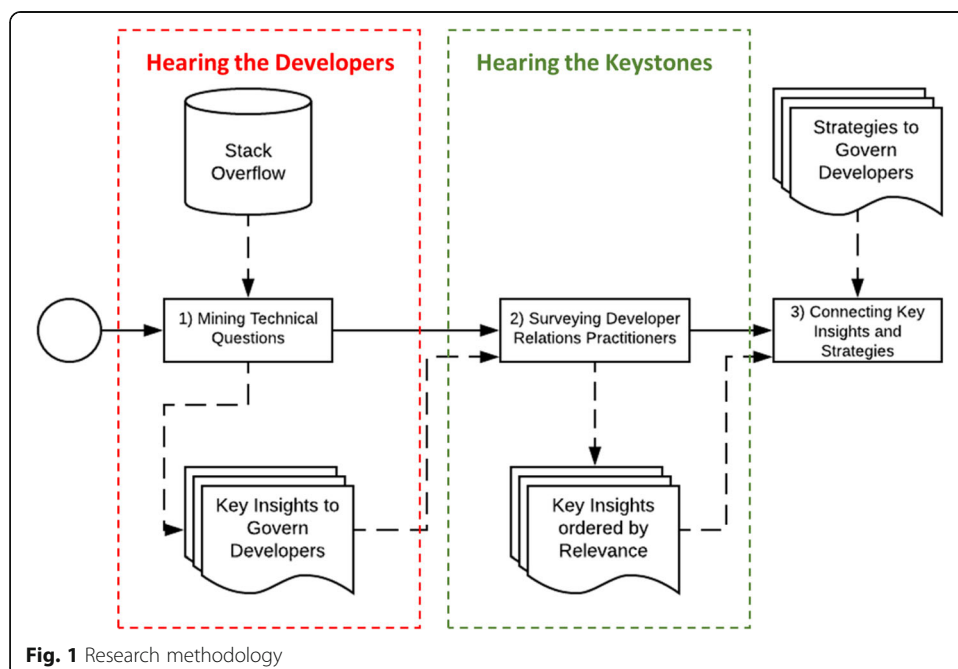


Fig. 1 Research methodology

researchers involved in the work. Section 6 covers the experimental design and discussion of the results of this study;

- 2) **Surveying Developer Relations Practitioners:** the key insights generated from the analysis of the results from the previous step are related to listening to the developer's perceptions within the MSECO. As a way to seek synergy, alignment between keystone's objectives and developers' expectations, a survey was conducted with professionals from the DevRel area. This survey examined the relevance of key insights to developer governance. Section 7 contains the description of the study and the analysis of the results; and
- 3) **Connecting Key Insights and Strategies:** after the result of the previous study, we made the association of the key insights with a set of strategies extracted from a systematic mapping study on developer governance in ecosystems. This allowed us to indicate concrete actions for MSECO governance. Section 7 also covers this study.

6 Mining technical questions

6.1 Study planning and design

Our research questions (RQ) are based on the principle "Representation" within the community governance (O'Mahony, 2007). The Representation principle means contributing members can be represented by community decisions or questions. The representation can be examined by the degree to which members can exercise voice on community members. Our questions and experimental design were inspired by the studies of Bajaj & Mesbah (2016) (focus: web developers), Barua et al. (2014) (focus: testing), and Rosen and Shihab (2016) (focus: mobile developers).

6.2 Study's goal and research questions

As a way to support the main research question "What can be understood from the three main MSECOs (i.e. Android, iOS, and Windows Phone) based on technical questions at Stack Overflow?", we defined a set of sub-research questions. We are currently investigating detailed insights regarding how to identify and support MSECO developer governance mechanisms from Stack Overflow. Those insights are very important to come up with rich information to aid decision-making based on the huge amount of available data. Our RQs are described using the GQM (Basili et al., 2007) approach as follows:

GOAL 1: Analyze how developers' activity is evolving in relation to number of questions, number of answers, and response time. The activity intensity corresponds to the frequency to which questions and answers are posted, including average time for topic answering.

- **RQ1.** What is the developer activity intensity from MSECO data available in Stack Overflow?
 - **Rationale:** The answer for this RQ can help us to analyze how developers' activity is evolving in relation to number of questions, number of answers, and response time.
 - **Metrics:** Number of questions and growth function; Number of answers and growth function; and Most frequent tags in recent questions.
- **RQ2.** What are the hot-topics extracted from technical questions asked by MSECO developers?

- **Rationale:** The answer for this RQ can help us to get an overview of what topics are covered and whether there is any difference among the analyzed ecosystems.
- **Metrics:** Number of clusters with similar topics; and Name of topics.
- **RQ3.** What are the hot-topics extracted from “How” and “What” questions asked by MSECO developers?
 - **Rationale:** The answer for this RQ can help us to get an overview of what topics are covered in how to perform development tasks and what aspects must be clarified by developers.
 - **Metrics:** Number of clusters with similar topics; and Name of topics.
- **RQ4.** What are the hot-topics extracted from unanswered technical questions asked by MSECO developers?
 - **Rationale:** The answer for this RQ can help us to get an overview of what topics are covered by unanswered questions and whether there is any difference among the analyzed ecosystems.
 - **Metrics:** Number of clusters with similar topics; and Name of topics.
- GOAL 2:** Analyze the developer engagement.
- **RQ5.** What are the platforms’ questions on which developers are more engaged?
 - **Rationale:** The answer for this RQ can help us to understand how much involvement in certain topics contributes to explore knowledge flow within an MSECO. As such, we can identify the most committed developers based on their most commented/ viewed questions.
 - **Metrics:** Number of Answers; Views count; Number of clusters; and Name of topics.
- **RQ6.** Is there any relation between questions and official events?
 - **Rationale:** We use time series to identify the frequent ecosystem questions in order to understand if topics have any relation with official events, such as platform launch. It is important to know how to analyze the effect of external events in the community.
 - **Metrics:** Posting frequency for 12 months.
- **RQ7.** What is the ranking of number of badges received by developers in each platform?
 - **Rationale:** The answer for this RQ can help us to obtain information on MSECO developers’ badges as well as to explore information about top developers.
 - **Metrics:** Number of badges for each MSECO; and More frequent badges for each MSECO.

6.3 Data selection

Stack Overflow makes its data publicly available in XML format licensed under CC BY-SA 3.0 license. For our purposes, we use *posts.xml*, which contains current posts’ text contents, as well as the answers/view count, tags, favorite count, and creation date. Our dataset contains information from Dec 01, 2017. Since the goal was to retrieve datasets from three MSECOs, we performed the mapping of tags that could represent Android, iOS, and Windows Phone MSECOs. This analysis allowed us to adopt the tags: *android*, *windows-phone*, and *ios*.

A total of 1,568,377 records from Stack Overflow related to MSECO were extracted to compose the dataset from January 2008 to December 2017, containing data related

to each MSECO: Android 62.9% (986,099), iOS 34.2% (535,876), and Windows Phone 2.9% (46,402). Part of the data obtained from each question dataset and an analysis of the available data are presented in this paper.

6.4 Study execution

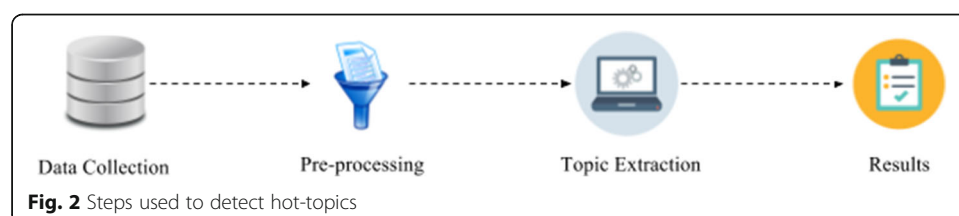
In the next subsections, we describe the procedures used to detect relevant topics, select the number of clusters and filter datasets to support the answering of RQs.

6.4.1 Detecting relevant topics

In order to answer some questions, we used LDA. To generate an LDA model, we need to understand how frequently each term occurs within each document. As such, we constructed a document-term matrix and our dictionary was converted into a bag-of-words (i.e. a common representation used in natural language processing and information retrieval). LDA was applied to each dataset related to the questions and silhouette method was used to evaluate the quality of clusters.

As the previous studies based on Stack Overflow mining, we used unsupervised approach to extract topics from its questions. Our methodology is composed of 4 steps (Fig. 2):

- **Data collection:** described in Section 6.2;
- **Pre-processing:** we pre-processed the textual content (Body) of the extracted posts in three steps. First, we discarded any code snippets that are present in the posts (i.e. enclosed in `<code>` HTML tags), because source code syntax (e.g. “if” statements and “for” loops) introduces noise into the analysis phase. Next, we removed all HTML tags (e.g. `<p>` and ``), since it is not the focus of our analysis. Third, we removed common English-language stop words such as “a”, “the” and “is”, which do not help to create meaningful topics. We used Spark as a framework that supports the analysis of big data. Data mining procedure was automated from the dataset construction to topic analysis. The pre-processing is step responsible for eliminating non-representative terms (e.g. stopwords, urls, emoticons and hashtags) in collection and make the feature extraction process. In this context, we use the NTLTK⁵ (Loper & Bird, 2002) like tool to eliminate non-representative terms and to make the feature extraction process, we use the bag-of-words approach with TF-IDF⁶ (Larson, 2010), where we eliminated terms with frequency less than 5;
- **Topic Extraction:** we used LDA (Krestel et al., 2009) since it is a statistical topic model used to automatically recover topics in several domains from a corpus of text documents. We chose LDA because it is able to model topics in large corpus; in



our case, the body of developers' questions related to MSECO. Moreover, we applied a partitioning technique called Silhouette (Rousseeuw, 1987) in order to identify the appropriate number of topics (and not a random choice). Such number provided by silhouette was applied as input to LDA. Each cluster is represented by a so-called silhouette, which is based on the comparison of its tightness and separation. This silhouette shows which objects lie well within their cluster, and which ones are merely somewhere else in between those clusters. The average silhouette width provides an evaluation of clustering validity. It might be used to select an appropriated number of clusters. Silhouette provides values in the range of -1 to 1 , where 1 means that the samples belonging to the cluster are far from the other clusters, 0 means that the division among the clusters is already at the edge of the separation, and -1 means that some samples have a chance to be assigned to the wrong cluster;

- **Results:** In the Section 6.5, for each research question presented in Section 6.2 questions, we analyze the results.

6.4.2 Filtering datasets

Aiming to answer RQ1, RQ5 and RQ6, we analyzed the full dataset. To answer RQ2, we filtered the original dataset to extract only the body of technical questions. As a way of answering the RQ3, we filter the original dataset to compose a dataset for "How" questions and another one for "What" questions. To do so, we searched the term "How" or "What" in the title of the questions and then we filtered the dataset. To create a dataset specific to RQ4 (unanswered questions) analysis, we extracted the questions with the answer count value equals to zero. For RQ7, we created a ranking of number of badges received by developers within the three MSECO. We used dataset information about 9795 developers from those MSECO ranked by reputation – number of conquered badges.

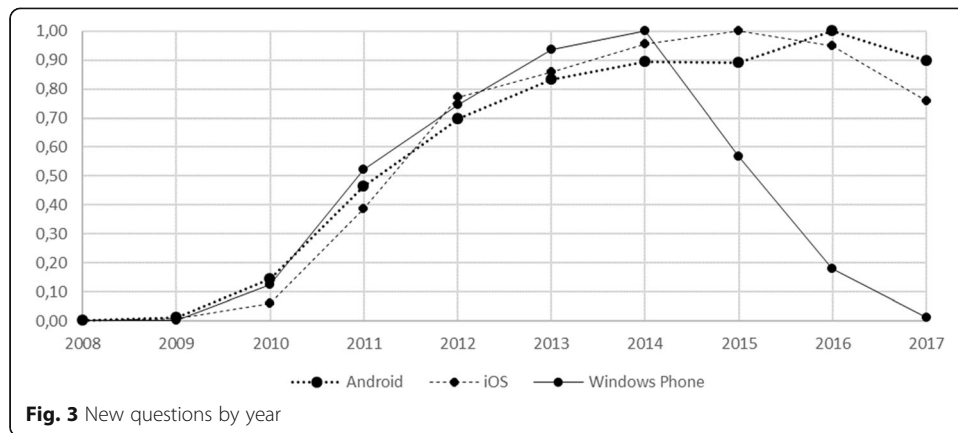
6.5 Results' analysis and discussion

At the end of each analysis in some RQs (RQ2, RQ3, RQ4 and RQ5), we present key insights as a set of notes that can guide researchers to use data to study MSECOs and that can be evaluated in further studies.

6.5.1 What is the developer activity intensity from MSECO data available in stack overflow?

Regarding the number of questions by year (2008 to 2017) (Fig. 3), the dataset allowed us to define a growth function for each MSECO as follows: Android: $a(x) = 16284x + 9045.8$; iOS: $i(x) = 9245.1x + 2739.4$; and Windows Phone: $w(x) = 358.25x + 2669.8$. The Android function $a(x)$ is 43% greater than iOS function and 97% greater than Windows Phone function $w(x)$. The iOS function $i(x)$ is 96% greater than $w(x)$.

We also analyzed the following null hypothesis "*There is no difference between the amounts of developers' posts among different MSECO*". The *Mann-Whitney test* was applied to verify the normality of the three samples with confidence level of 95%. We identified that the samples follow the normal distribution. There was a statistically significant difference among groups as determined by *one-way ANOVA*, $p = .001$. A *Tukey post hoc test* revealed that the amount of Windows Phone questions was statistically



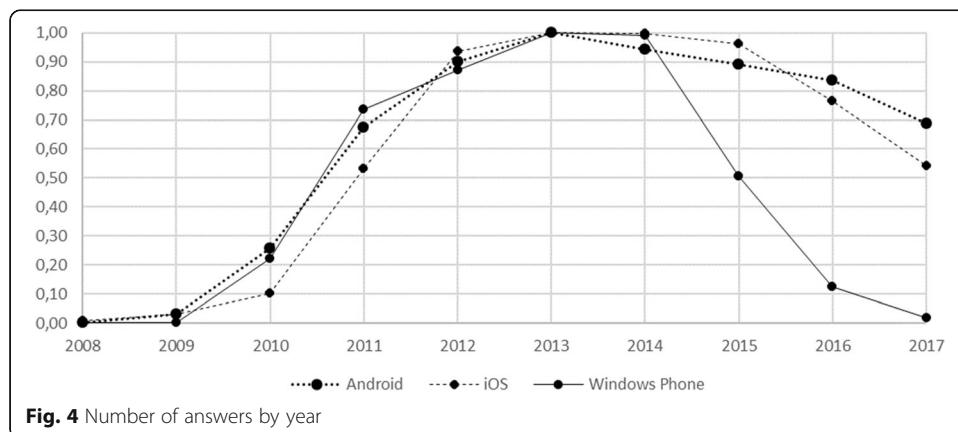
significant lower than Android ($93,969 \pm 22,638$, $p = .0096$). There was no statistically significant difference between the Android and iOS questions ($p = .134$), and between Windows Phone and iOS questions ($p = .096$). The Windows Phone began to be discontinued by Microsoft in 2015, which has affected the community involvement (Fig. 3). We can perceive that Android and iOS are the main MSECO in the market.

Regarding the number of answers by year (Fig. 4), the dataset also allowed us to define a growth function for each MSECO as follows: Android: $a(x) = 31061x + 3728$, 1; iOS: $i(x) = 19136x + 1017$, 7; and Windows Phone: $w(x) = 869,07x + 2462$, 6. Android function $a(x)$ is 38.4% greater than iOS function and 97.2% greater than Windows Phone function $w(x)$. iOS function $i(x)$ is 95.3% greater than $w(x)$.

The Windows Phone began to be discontinued by Microsoft in 2015, which has affected the community’s involvement in questions (Fig. 3) and answers (Fig. 4). With this information, we can perceive the Android and iOS as the main MSECOs in the market.

We also analyzed tags that represent questions that take more time to be answered and questions that are quickly answered by each MSECO. We ranked 500 questions that take more time to be answered and that are quickly answered (each question has information about related tags and time to answer).

Five Android tags with more than 500 questions that take more time to be answered are described next: *android-syncadapter* (26.4 h) – a service that synchronizes data



between an Android device and a server; ***android-espresso (18.5 h)*** – espresso is a library which is used to write Android UI (User Interface) tests; ***android-testing (18.3 h)*** – Android testing framework that provides an architecture and tools to test each level from unit to framework; ***google-drive-android-api (16.3 h)*** – Drive Android API is a native API simplifying many common associated tasks using Drive service in mobile devices; and ***android-source (15.5 h)*** – questions about source code and related themes: how to contribute and/or porting etc.

Five Android tags with more than 500 questions that are quickly answered are described next: ***android-context (15 min)*** – interface to global information about an app environment. It allows access to up-calls for app-level operations such as launching activities, broadcasting and receiving intents etc.; ***android-button (16.5 min)*** – this tag is for questions about Buttons over Android platform; ***android-alertdialog (17 min)*** – a subclass of Dialog that can display one, two or three buttons; ***android-asyncTask (19 min)*** – AsyncTask enables proper, easy use of UI thread. AsyncTasks should be ideally used for short operations (few seconds, at most); and ***android-textview (19 min)*** – Android user interface component that displays text to the user.

Five iOS tags with more than 500 questions that take more time to be answered are described next: ***ios-app-extension (59.2 h)*** – a feature introduced in iOS 8 that was created to perform a specific task, such as to enable Safari pages sharing through an app, or to display an app interface in Notification Center; ***ios-ui-automation (47.4 h)*** – this tag specifically focuses on using this functionality in the iOS development, and questions related to scripts can be used to automate interaction between user and app; ***facebook-ios-sdk (12.9 h)*** – Facebook’s SDK for developing Facebook-connected apps for iOS devices; ***google-maps-sdk-ios (9.5 h)*** – Google Maps SDK for iOS allows users to view and interact with a Google map; and ***ios10 (5.9 h)*** – iOS 10 is the tenth version of Apple’s iOS mobile operating system.

Five iOS tags with more than 500 questions that are faster to be answered are described next: ***ios4 (35 min)*** – iOS 4 was made publicly available for iPhone and iPod Touch on June 21, 2010. It has been succeeded by ios5 (tag ***ios5-48 min***) which was released on October 12, 2011; ***ios-autolayout (1 h)*** – auto Layout dynamically calculates the size and position of all the views in a view hierarchy, based on constraints placed on those views; ***ios-provisioning (1 h)*** – the process of preparing an app to run on an iOS device; and ***ios6 (1.17 h)*** – related to iOS platform that provides more than 200 new features, including a new Maps app, Siri updates, Siri for iPad (3rd generation) etc.

Five Windows Phone tags with more than 500 questions that take more time to be answered are described next: ***windows-phone-voip (53.4 h)*** – related to Windows Phone integration with phone services, and the ability to retrieve incoming VoIP calls in the background using push messaging; ***windows-phone-emulator (16.3 h)*** – Windows Phone Emulator presents Windows Phone Interface on an Windows PC; ***windows-phone-silverlight (8.5 h)*** – Microsoft Silverlight is a free web-browser plug-in that enables interactive media experiences and rich business apps; and ***windows-phone-8-emulator (8.3 h)*** and ***windows-phone-7-emulator (7.8 h)*** – the emulator allows for the development and testing of Windows Phone 7.x and 8 apps out of a hardware device.

Five Windows Phone tags with more than 500 questions that are faster to be answered are related to Windows Phone platforms versions: ***windows-phone-7 (1 h)***,

windows-phone-7.8 (1.1 h), windows-phone-8 (2 h), windows-phone-7.1 (2.2 h), and windows-phone-7.1.1 (3.6 h).

Using a simple word count algorithm, we extracted a list of each platform containing the top 30 tags that were repeated in the recent questions as shown in Table 1. In general, it is possible to realize that the five most repeated tags in the Android MSECO is related to the programming language (Java), using Android Studio (Studio), with elements of user interface (Layout and Fragment) and back-end infrastructure for mobile applications (Firebase). In the iOS MSECO, the most used tags are related to: programming languages (Swift) and Objective-C (Object), use of the development environment (XCode), iPhone usage procedures for debug and deploy mobile applications, and user interface elements (*UITableView*) regarding the most common type of views used in iOS apps. Recent questions in the Windows Phone MSECO have the following most commonly used tags: relationship to application interface code programming (XAML), the language behind the visual presentation in Window Phone apps, app behavior programming (Silverlight), use and configuration of the Visual (Studio) development environment, Framework (Net) that contains packages to support an intuitive development process, and the settings of the *App.xaml* – a declarative start point of an app.

It is interesting to observe that there is an intersection between topics pertaining to each MSECO, there are common points involving: programming language adopted in the platform, development environment, user interface (UI) programming, coding styles, and development support infrastructure from apps.

Table 1 The Most 30 Repeated Tags in Recent Questions

Android		iOS		Windows Phone	
Tag	#	Tag	#	Tag	#
Java	10,259	Swift	24,979	Xaml	6841
Studio	5016	Object	9820	Silverlight	5062
Firebase	3664	Xcode	8366	Studio	2623
Layout	2422	Iphone	3990	Net	2564
Fragment	2322	Uitableview	3155	App	1516
Gradle	1894	Firebase	3049	Runtime	1365
Xamarin	1611	Core	1750	Wpf	1291
Listview	1467	Android	1579	Mvvm	1245
Recyclerview	1419	Notification	1387	Cordova	1229
Xml	1313	Google	1327	Mobile	988
Api	1313	Xamarin	1165	Universal	938
Sqlite	1291	Push	1110	Bind	925
Json	1238	Data	1082	Listbox	855
Map	1217	Cordova	1077	Xml	850
Service	1200	UICollectionVIEW	1049	Control	820
Database	1187	Facebook	1036	Json	805
Cordova	1138	Javascript	988	Xna	782
Intent	947	App	960	Store	777
Io	934	Autolayout	896	Image	773
Notification	933	Uiview	839	Visual	770

6.5.2 What are the hot-topics extracted from technical questions asked by MSECO developers?

Regarding all the questions related to MSECO, the number of topics and silhouette value were obtained through the silhouette method for each MSECO as follows: Android – $n = 4$ (0.64), iOS – $n = 3$ (0.86), and Windows Phone – $n = 3$ (0.87). We used the number of topics as input to LDA algorithm. Table 2 shows the results.

In the Android MSECO, questions related to **Project topic** involve the basic of Android projects, such as starting new projects, importing/exporting projects, and creating/manipulating activities. **User Interface topic** covers questions about placement, alignment and justification of objects with respect to a container element. Questions related to **Exceptions topic** covers issues related to a condition that requires deviation from the Android program's normal flow. Finally, **Notifications topic** covers technical questions related to a user interface element that a developer can display outside the app's normal UI to indicate that an event has occurred. Users can choose to visualize the notification while using other apps and respond to it according to their convenience.

Regarding the iOS MSECO, the **Data Binding topic** covers mechanisms used to synchronize an UI with an underlying data model. **User Interface topic** in iOS covers the user interface control, and adaptation to any size changes. **Project topic** involves similar questions within Android community with focus in the build of apps to hardware devices.

Questions in the Windows Phone MSECO have the following topics: **Services** – that involves web and data services that use an open XML-based language to describe their web-based API; **Data Binding** – a connection/binding between UI and a data object allows data flow between such tiers; and **Frameworks** – questions referring to dynamic-link libraries, frameworks to support game development, native functionalities of the system etc.

There are common points in the intersection between topics of different MSECO: data binding mechanisms, user interface (UI) programming, and development support infrastructure. This leads us to the following key insight:

Key Insight #1: *The most commonly used tags in recently added questions may indicate the most frequent barriers faced by developers willing to participate in an MSECO. This scenario can serve as a monitoring strategy to support a keystone in recruiting and educating developers.*

Table 2 Extracted topics from all questions

Android	Project	file, project, activity, class
	User Interface	activity, view, layout, xml
	Notifications	service, device, data, notification,
	Exceptions	androidruntime, lang, thread, method
iOS	User Interface	cell, table, tableview, uitableview
	Data Binding	view, data, controller, screen
	Project	xcode, project, device, iphone
WP	Services	wSDL, keys, resx, reminder
	Data Binding	page, data, xaml, control
	Frameworks	xna, dll, native, reflection

6.5.3 What are the hot-topics extracted from “how” and “what” questions asked by MSECO developers?

Regarding the whole set of “How” questions related to the different MSECO, the number of topics and silhouette value were obtained through the silhouette method for each MSECO: Android – $n = 4$ (0.39), iOS – $n = 4$ (0.40), and Windows Phone – $n = 3$ (0.45). We used the number of topics as input to LDA algorithm. Table 3 shows the results.

The “How” questions are related to the steps to perform some task during the app development. In Android MSECO, topics cover the game development, app deployment over a real device, operation of authentication services such as Facebook and email, and, finally, behavior of elements that make up the app screen. In iOS MSECO, topics involve coding of controllers for reuse of component interface, configuration of permissions and verification within the app, design patterns aspects, and conversion of interface component values to string for manipulation of value informed by a user within the app. In Windows Phone MSECO, developers have questions on how to perform the steps to use notification service, use of sensors debugging mechanisms, and data persistence with databases.

Regarding all the “What” questions related to the different MSECO, the number of topics and silhouette value were obtained through the silhouette method for each MSECO: Android – $n = 3$ (0.40), iOS – $n = 2$ (0.62), and Windows Phone – $n = 3$ (0.46). We used the number of topics as input to LDA algorithm. Table 4 shows the results. “What” questions are related to the need for development knowledge on how something works during app development. In Android MSECO, we realized the need for knowledge about interoperability, i.e. interaction between an app and other apps available or unavailable on the user’s device. Persistence and Database involve decisions about what solution (e.g. tool, data format, and code) should be used in a specific scenario. In iOS MSECO, a topic covers errors while debugging parts of the app. There is also a specific interest in understanding terms related to audio conversion within an app. In Windows Phone MSECO, the first topic (Silverlight) is related to the framework used for the development of Windows Phone 7.0 applications covering help in methods, SDKs, and development. Another topic is related to understanding what the

Table 3 Hot-topics in “How” questions

Android	Game	Game, country, multiplayer, uri, basic, slashes, escaping, alarm, layered, appdata
	Deployment	Sdk, device, studio, manager, build, nexus, adb, screen, windows, format
	Authentication	Facebook, email, token, login, algorithm, callback, requirement, user, integrated, quickstart
	UI	Code, show, button, activity, xml, image, file, layout, screen, new
iOS	UI	Show, button, xcode, image, user, controller, need, bar, screen, iphone
	Permissions	Peer, bitcode, sd, dns, identifier, cart, performance, captivenetwork, wifi, unlock
	Design Pattern	Class, type, object, protocol, testers, net, utils, variable, declared, predicate
	Type Conversion	String, array, picker, slider, datepicker, method, convert, nsstring, facebook, NSMutableArray
WP	Notifications	Text, push, file, notification, package, memory, uri, notifications, font, device
	Sensor Debug	Null, debug, string, key, getslotfrombufferlocked, surfacetextureclient, gyroscope, uncalibrated, value, drift
	Database	Java, strictmode, activitythread, method, support, sqllitedatabase, methods, msg, database, loop

Table 4 Hot-topics in “What” questions

Android	Interoperability	Fragment, activity, android, whatsapp, image, data, user, device, method, time
	Database	Java, strictmode, activitythread, method, support, sqlitedatabase, methods, msg, database, loop
	Persistence	Code, release, boot, toolbar, kernel, clock, loginactivity, invisible, preference, loader, sprites
iOS	Debugging	Code, view, file, one, xcode, user, new, method, error, image, time
	Audio Conversion	Bin, armv7, kaudioformatmpeg4aac, detailsviewController, armv7s, frequency, distributions, watchkitextension, frontviewController, rotation
WP	Silverlight	Know, need, help, going, Silverlight, sdk, one, development, wp7, method
	UI Behavior	Page, background, bar, time, using, back, first, theme, wp8, information
	Azure Services	Uri, data, user, notification, service, azure, well, build, code, push

app’s default UI behavior for Windows Phone is. Finally, what Azure Services is and how the service work are other topics.

6.5.4 What are the hot-topics extracted from unanswered technical questions asked by MSECO developers?

Regarding the most visualized unanswered developer questions in Stack Overflow, the number of topics and silhouette value were obtained through the silhouette method for each MSECO: Android – $n = 2$ (0.45), iOS – $n = 2$ (0.58), and Windows Phone – $n = 3$ (0.47). We used the number of topics as input to LDA algorithm. Results are shown in Table 5.

Our goal in answering this question (RQ4) is to explore the most frequent questions the community points out but at the same time does not hold the knowledge. On the other hand, questions may be very obvious to the community, or they may already have answers and they have not been moderated. In Android MSECO, the most frequently unanswered questions are related to: **Deployment Issue** – issues detected in code that influence app installation, debugging, and testing over a device; and **Hidden Menu** – involves activating the menu with factory commands that provide a large amount of information about device hardware and system since an average user might have difficult to access.

In iOS MSECO, the first topic (**Facebook Login Error**) refers to the difficulty of login into Facebook within Safari browser and feedback of information into an app.

Table 5 Hot-topics in “Unanswered” questions

Android	Deployment Issues	Code, using, file, problem, android, device, error, one, help, image
	Hidden Menu	Android, java, rtn, sensor, 11, 12, 22, 05, com, float
iOS	Facebook Login Error	Facebook, user, using, safari, device, login, notification, keyboard, code, server
	Video	View, code, using, file, error, problem, set, controller, works, video
WP	Analytics Integration	List, item, column, listpicker, pull, textblock, database, name, localytics, radius
	Libraries Issues	System, dll, exception, Microsoft, windows, speech, error, threading, mscorlib, task
	Design Tool	Expression, remoting, designhost, isolation, tribe, vimeo, score, silverlightplatform, Microsoft, canceltoken

This is related to the Facebook's authentication mechanisms that are used within mobile apps. Another topic covers issues with manipulation of **video** components. In Windows Phone MSECO, the topic **Analytics Integration** refers to the use of analytics SDKs within app components as a way to map user behavior in a more detailed way. The topic **Libraries Issues** covers exceptions and other errors in the use of libraries; their interaction with Windows Phone can be hard due to the frequent exchange of operating system versions, causing several problems of library incompatibility. The topic **Design Tool** is related to problems that occurred during the use of a tool not only for designing but also for developing an app.

From the analysis of those MSECO, we observed that even a moderation of the issues does not avoid problems that are frequently informed throughout the app development for MSECO. Our analysis made for this point took into account the most visualized issues so that the topics may be related to problems regularly faced by developers. MSECO topics share as a common characteristic the fact that they are terms related to those who have already advanced in development: deployment, analytics, service authentication, libraries, and design.

Analyzing the previous research questions shows that there seems to be little intersection of interests. This may be evidence that MSECOs are different in terms of "interests, challenges, difficulties". This may indicate that the strategies actually vary greatly, which refers to the need for researching the commonalities as a way of establishing a general model of developer interactions and their governance within MSECOs. Another point to be explored is how this lack of intersection of interests drives exchange of information between developers working on more than one MSECO. The little intersection shows that the strategies taken at a managerial level may affect the interactions and motivations of the developers.

6.5.5 What are the platforms' questions on which developers are more engaged?

In order to analyze this research question, we defined two perspectives: 1) to analyze engagement by number of answers of developers who participate in MSECO and Stack Overflow; and 2) analyze the engagement by number of question visualizations, i.e. developers who do not participate in Stack Overflow but visualize questions and add answers. Table 6 shows five records with the most popular answered questions obtained from each platform dataset, and Table 7 presents the most visualized answers by developers. For each case (NumAnswers and ViewCount, respectively), we used LDA method to identify in which topics developers are more engaged. In turn, the amount of topic clusters was defined using the silhouette algorithm.

Regarding developer engagement in Stack Overflow and the most popular answered questions, the number of topics and silhouette value were obtained through the silhouette method for each MSECO: Android – $n = 4$ (0.64), iOS – $n = 4$ (0.86), and Windows Phone – $n = 3$ (0.87). We used the number of topics as input to LDA algorithm. Results are shown in Table 8.

In Android MSECO, the most frequently answered topics were related to: **Data Binding** – to write declarative layouts and minimize glue code necessary to bind app logic and layouts; **IDE** – use code editing, debugging and performance tools; **User Interface** – to create a dynamic and multi-pane user interface to encapsulate UI components and activity behaviors

Table 6 Questions ordered by number of answers

Android	R cannot be resolved - Android error Why is Android emulator so slow? How can we speed up the Android emulator? "Conversion to Dalvik format failed with error 1" on external JAR Developing for Android in Eclipse: Rjava not regenerating Close/hide Android Soft Keyboard
iOS	How to make an UITextField move up when keyboard is present? What does this mean? "NSUnknownKeyException", reason: ... this class is not key value coding-compliant for the key X" Placeholder in UITextView Applications are expected to have a root view controller at the end of application launch How to change Status Bar text color in iOS 7
WP	Windows Phone 7 closes application Windows Phone 8 emulator can't connect to the internet Resources for Windows Phone 7 development Unable to create the virtual machine Windows Phone 8 Emulator not launching. Error code 0x80131500

into modules of activities; and **Back-end infrastructure** – use a platform that helps to grow the user base and monetize the app.

In Windows Phone MSECO, the topics were related to: **Event Handler** – the use of handling manipulation events methods for processing touch input; **IDE** – tools to support app development, including emulators and migration tools (WP7.X to WP8.X); and **User Interface control** guidelines.

In iOS MSECO, some topics were the same of the others. Additional topics were: **User Interface; Data Binding; Notification Services** – local and push notifications for keeping users informed with relevant content, whether the app is running in the

Table 7 Questions ordered by number of views

Android	Can't start Eclipse - Java was started but returned exit code = 13 Get screen dimensions in pixels Close/hide Android Soft Keyboard What is the difference between "px", "dp", "dip" and "sp" on Android? R cannot be resolved - Android error
iOS	How can I develop for iPhone using a Windows development machine? What does this mean? "NSUnknownKeyException", reason: ... this class is not key value coding-compliant for the key X" How to make an UITextField move up when keyboard is present? How to change Status Bar text color in iOS 7 Vertically align text to top within a UILabel
WP	.Net - DateTime.ToString("MM/dd/yyyy HH:mm:ss.fff") resulted in something like "09/14/2013 07.20.31.371" Install Visual Studio 2013 on Windows 7 Can we install Android OS on any Windows Phone and vice versa, and same with iPhone and vice versa? How to Install Windows Phone 8 SDK on Windows 7? How to trigger event when a variable's value is changed?

Table 8 Hot-topics – the most answered questions

	Topic	Words
Android	Data Binding	Android, Text, Java, API
	IDE	Studio, File, Adb, Device
	User Interface	Activity, Fragment, View, Image
	Back-end Infrastructure	Firebase, Data, Notification, Time
iOS	User Interface	Swift, View, Change, Image
	Data Binding	Data, View, Control, Swift
	Notification services	Can, Call, Work, Notification
	Programming Language	Swift, Object, Xcode, Value
WP	Event Handler	Event, Develop, Silverlight, Visual
	IDE	WP7, file, service, emulator
	User Interface	Listbox, control, page, image

background, or inactive; and *Programming Language* – the use of Swift and Objective-C in XCode to develop apps.

Regarding the engagement from the questions that are most visualized in Stack Overflow, the number of topics and silhouette value (obtained by the silhouette method) respectively for each MSECO are: Android – $n = 4$ (0.62), iOS – $n = 3$ (0.81), and Windows Phone – $n = 4$ (0.75). We used the values indicated for the number of topics. Results are shown in Table 9.

In Windows Phone MSECO, developers work to upgrade their apps to the newest platform as a way to support new features (e.g. sensor data). An Android developer must decide whether to build a single app or multiple versions to run on top of the broad range of devices by the use of fragments. In iOS MSECO, the most visualized questions refer to the use of integrated development environment, main programming, and design/development of user interface.

The most visualized topics can indicate frequent barriers faced by app developers because those questions can be found by any developer using a search engine as

Table 9 Hot-topics – the most visualized questions

	Topic	Words
Android	IDE	Device, android, java, string
	User Interface	View, set, button, listview
	Basic Steps	File, create, project, use
	Interface behavior	Activity, fragment, call, service
WP	Platform	App, 8, 1, T, work
	Notification Services	Data, call, service, notification, method
	Data Binding	Image, item, bind, listbox, control
	Page Navigation	File, page, button, navigation, wp7
iOS	IDE	Xcode, Imag, View, iPhone, Chang
	Programming Language	Object, Swift, c, text, io
	User Interface	Uitableview, view, uiview, cell, anim

Google, for example. The analysis of engagement from the perspective of the most commented/visualized questions allowed us to define the following key insight:

Key Insight #2: *The most visualized topics as well as the topics in which developers are most committed to respond can indicate a community of experts who can help to reduce frequent barriers to participation in MSECO.*

6.5.6 (RQ6) is there any relation between questions and official events?

In order to answer this question, we selected a period between February/2015 and January/2016 since it covers official announcements of the MSECO organizations' official channels (Google I/O, WWDC Apple Developer, and Microsoft Build, which includes the latest edition of Microsoft Build covering Windows Phone aspects). The first analysis allowed us to verify whether there was a similar behavior in the posting frequency among the different MSECO (Table 10). For statistical analysis, data were normalized to a range [0, 1]. We calculated the posting frequency for each day of the year and then we divided each element by the maximum element. Finally, we calculated the average for each month.

We analyzed the following null hypothesis "There is no difference between the frequencies of developers' posts among MSECOs in a selected period of time". The selected period was between February/15 and January/16. The *Mann-Whitney test* was applied to verify normality of the three samples with confidence level of 95%. We identified that the samples follow the normal distribution. There was a statistically significant difference between groups as determined by *one-way ANOVA*, $p = .0001$. A *Tukey post hoc test* revealed that the frequency to which Windows Phone developers post questions was statistically significant lower than iOS developers ($.334 \pm .018$, $p = .0001$) and Android developers ($.365 \pm .018$, $p = .0001$). There was no statistically significant difference between Android and iOS developers ($p = .233$).

From Table 10, we can identify seasonal points within the time series formed by the posting frequency. The highest point for Windows Phone within the studied period was the first month (February/2015); it was the last month of the series for Android-related posts (January/2016); and it was the sixth month for iOS (July/2015).

Table 10 Posting frequency in a specific year (Feb/2015 – Jan/2016)

Month/Year	Windows Phone	Android	iOS
February/15	0.43 ^	0.62	0.59
March/15	0.34	0.71	0.66
April/15	0.39	0.72	0.67
May/15	0.37	0.70	0.65
June/15	0.35	0.69	0.67
July/15	0.39	0.74	0.73 ^
August/15	0.32	0.71	0.70
September/15	0.31	0.67	0.71
October/15	0.22	0.70	0.69
November/15	0.33	0.67	0.62
December/15	0.27	0.72	0.66
January/16	0.32	0.77 ^	0.70

In February/2015, Microsoft announced improvements for Windows Phone developers: Windows Phone download and in-app purchase reports have been optimized to deliver information faster. Microsoft also announced a Windows App Studio Beta bringing new features such as a full-featured logo and an image wizard with built-in image controls and conversion tools, besides improved Facebook and YouTube DataSources matching their latest API releases. To analyze the impact of “in-app purchase”, we found that the questions in February/2015 are related to: code recovery error in a WP8.0 platform app; expectation when trying to add coins to the current balance; when using a method, a developer cannot find item in catalog; developers use a method that is deprecated in new release; and once the purchase has been done, code does not run at all. The filter for “Windows App Studio” related questions did not return information. This issue may be related to the fact that App Studio is an online app creation service.

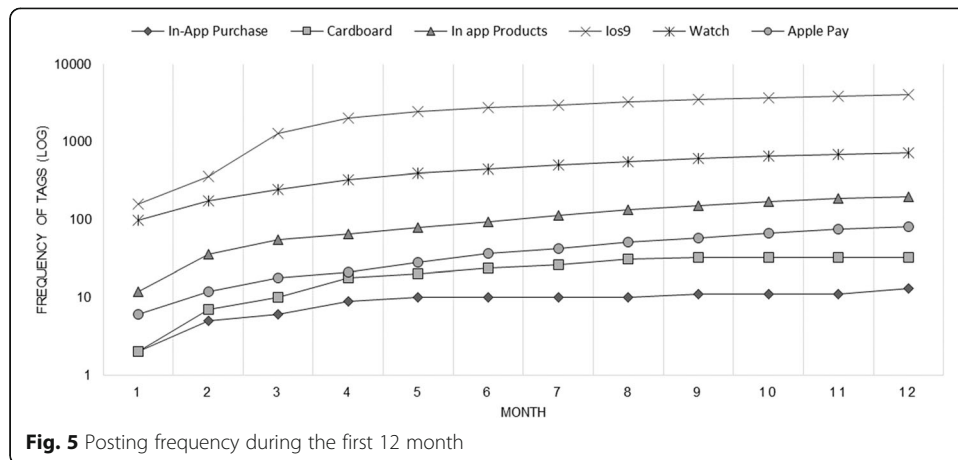
In January/2016, Google announced new features to better understand player behavior with Player Analytics, inclusion of promo codes for apps and in-app products in the Google Play Developer Console. Another announcement was the Cardboard SDKs for Unity and Android support spatial audio in order to help developers in creating equally immersive audio experiences in a virtual reality (VR) app. Regarding the “Cardboard”, developers have published questions involving: creating a stereo 360 player by using cardboard SDK for unity; application shows a small screen on Android device; error in cardboard SDK; and errors when using a demo. When the term is “in-app” products, developers describe questions related to: billing failed; allowing to buy a “zero price product”; user interface behavior; techniques for implementing in-app; use of Cordova/PhoneGap; and testing in-app purchase does not work.

In iOS MSECO (July/2015), announcements englobed: Apple Previews iOS 9, News App for iPhone & iPad, OS X El Capitan, New Apple Watch Software watchOS 2 (Native Third-Party Apps, New Watch Faces & Enhanced Communications Features), and Expanding Benefits with Merchant Rewards & Store Cards (Apple Pay). With the announcement of iOS 9, developers have posted questions related to: XCode 7, Swift 2.0 and interface settings, which involves updating SDKs, font-rendering crashes, failures when trying to launch emulators, use of TouchID, and deprecated methods. In the case of Apple Watch, questions relate to the implementation of features, use of gestures, testing the XCode emulator, user interface, and how to use the sensors. While using Apple Pay, developers questioned crashes involving Swift Apple Pay, how to use Apple Pay with a PayPal SDK (BrainTree), and integration with Apple Passport.

Figure 5 shows that the use of tags related to announcements maintains an accumulated growth of questions until the fourth month. After that, the behavior stays almost constant based on the difference between the last month and the current one. From the analysis, we can perceive that technical questions emerge when a keystone delivers new technologies; a keystone must effectively deliver new technologies, processes or ideas to the ecosystem’s participants. The analysis led us to the following key insight:

Key Insight #3: *Questions posted in Stack Overflow next to official MSECO announcement periods can help a keystone to manage strategies to add new MSECO resources (e.g. platforms, SDKs, APIs, programming languages). When such new technologies are released to the market, a keystone should be able to manage them easily.*

From this key insight, we can perceive a difference between IT governance – in which business strategies are not necessarily reflected in the IT decisions (Manikas et al.,



2015) – and MSECO – in which business strategies affect the developers’ communities (e.g. APIs and SDKs announcements).

6.5.7 What is the ranking of number of badges received by developers of each platform?

For this research question, we created a ranking of number of badges received by developers within the three MSECO. We used dataset information about 9795 developers from those MSECO ranked by reputation – number of conquered badges. Table 11 shows this ranking. Due to statistical analysis purposes, we normalized the data following the procedures adopted in the previous research question.

We investigated 9795 developers with badges based on the following hypothesis: “There is no difference between the numbers of badges received by developers from the different MSECO”. Applying *One-way ANOVA test*, we perceived that the significance value is $p = .0001$, which is below 0.05. Therefore, there is a statistically significant difference in the number of badges among the different MSECO. A *Tukey post hoc test* revealed that the number of badges acquired by Windows Phone developers was statistically significant lower than iOS developers ($.00052 \pm .00019, p = .015$) and Android developers ($.00089 \pm .00019, p = .0001$). There was no statistically significant difference between Android and iOS groups ($p = .139$).

Table 11 Ranking – number of conquered badges

Ranking	NumBadges (Android)	NumBadges (iOS)	NumBadges (Windows Phone)
1°	14,779	14,779	14,779
2°	5576	4074	5166
3°	4202	3093	4202
4°	4074	2588	3093
5°	3093	2130	3035
6°	3035	2062	2755
7°	2588	1865	2732
8°	2421	1650	2421
9°	2150	1623	2279
10°	2062	1621	2130

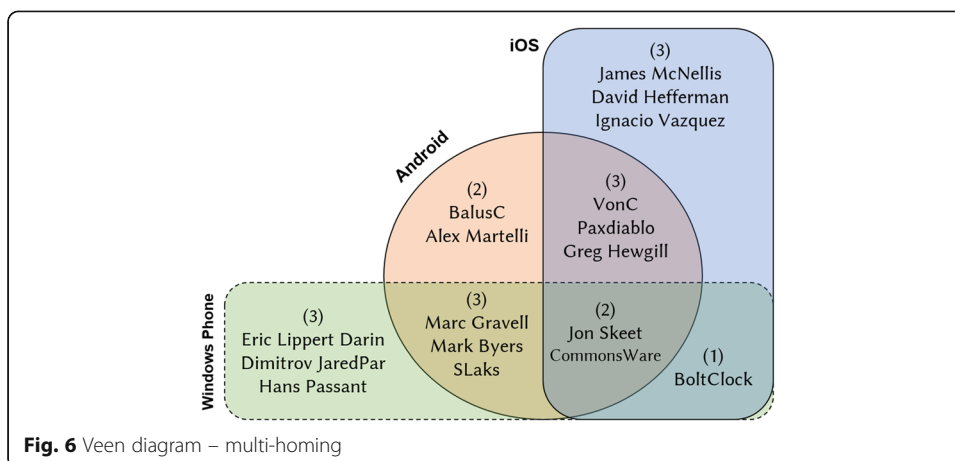


Fig. 6 Veen diagram – multi-homing

From the first ten developers in the ranking, it was possible to identify that some of them act as multi-homing, i.e. they play in more than one MSECO, helping to answer questions and manage communities in Stack Overflow. For example, see the first one: Jon Skeet (Fig. 6). From the ecosystem perspective, such developer profile is important because it fosters the exchange of knowledge acquired from the interactions among ecosystems and developers. We also analyzed the badges conquered by developers in each MSECO and created a ranking with the five most frequent badges as shown in Table 12. The Mortarboard badge is the only one present in Android (1°), iOS (1°) and Windows Phone (3°). It is a bronze *participation* badge earned when developers conquer at least 200 reputation points in a single day (200 is the daily maximum).

Analyzing Android MSECO, second badge in the ranking is *Multithread*, i.e. a participation badge earned when at least 400 total score for at least 80 non-community wiki answers is conquered. The third badge, *Legendary*, is a gold participation badge earned when 200 daily reputation is conquered 150 times. In turn, *Quorum* is a bronze participation badge earned when a developer reaches one post with score of two on Meta Stack Exchange (i.e. part of the site where users discuss Stack Overflow workings and policies). Finally, *Great Answer* is a gold answer badge when an answer’s score of 100 or more is conquered. The five most frequent badges in Android MSECO are participation-related and one is focused on answers.

For reviews, we have expanded the filter for a period of an extra month because some announcements occurred at the end of such indicated month. In iOS MSECO, *Reviewer* is a silver moderation badge earned when the developer complete at least 250 review tasks. Next, *Great Answer* is a gold answer badge as explained above. The

Table 12 Top Five Badges earned in each MSECO

Ranking	Android	iOS	Windows Phone
1°	Mortarboard	Mortarboard	Enthusiast
2°	Multithread	Reviewer	Good Answer
3°	Legendary	Great Answer	Mortarboard
4°	Quorum	Editor	Talkative
5°	Great Answer	Cleanup	Excavator

fourth badge, *Editor*, is a bronze moderation badge conquered when the developer make some editions for the first time. Finally, *Cleanup* is also a bronze moderation badge when the developer made his/her first rollback. The five most earned badges in MSECO iOS are related to moderation and one of them refers to answers.

In Windows Phone MSECO, the first most frequent badge is *Enthusiast*, i.e. a silver participation badge earned when a developer visits Stack Overflow every day for 30 consecutive days. The second badge is *Good Answer*, i.e. a silver answer badge earned when an answer gets a score of 25 or more. *Talkative* badge is part of participation category – this badge is earned when the developer posts ten message evaluated with one or more stars. The last badge is *Excavator*, i.e. a bronze moderation badge earned when a developer edit a first post that was inactive for six months. The five most frequent badges in MSECO Windows Phone are related to participation, response, and moderation.

The last proposition refers to the identification of developers and technical communities within Stack Overflow – it can play as “an extension” of the keystone role. This extension rises from the technical knowledge flow and community control:

Key Insight #4: *Badges can help a keystone to manage strategies related to technical resource exploration, active developer in the community, and community control by fostering relationships with top developers in the ecosystem.*

6.6 Threats to validity

Below we present the possible threats to validity involved in this study, and how we mitigated it.

Constructo validity: the theoretical basis of this study considered the weaknesses pointed out in recent literature reviews published in the SECO field, i.e. in-depth studies. The choice for Stack Overflow as a Q&A repository is due to the presence of developers who also post questions and answers related to the mobile platform domain.

Internal validity: datasets were not selected randomly, but they were related to the studied MSECO. To reduce the effect of the experimenters’ expectation, the study’s analyses followed the procedures indicated by algorithms or statistical analyses.

External validity: the environment is not different from the real one since Stack Overflow is a repository with questions from developers who are somehow participating in an MSECO. In addition, our analysis considered the three main MSECO in the market: Android, iOS, and Windows Phone.

Conclusion validity: The statistical analyses and/or result interpretation were based on algorithms for topic extraction (LDA), word counting, and procedures for hypothesis testing with a confidence level of 95%.

7 Key insights and strategies

After “listening” the voice of developers by mining technical questions from Stack Overflow, we proposed a set of four key insights related to developer governance in MSECO. Considering the higher level of key insights, we investigated them regarding how MSECO developers’ governance mechanisms can be identified and supported by Q&A repositories from practitioners’ perspective. From those insights, a set of strategies was proposed in this study but the relevance of those insights should be firstly evaluated with professionals who work/worked governing developers in MSECO.

7.1 Study planning and design

7.1.1 Study's goal and research questions

This section presents a survey planned and executed **with the goal of** analyze the four key insights extracted from mining mobile application related questions in Stack Overflow **with the purpose of** characterizing **with respect to** their relevance **from the point of view of** practitioners **in the context** of developer governance in MSECO.

7.1.2 Participants' selection

To analyze the set of key insights, we contacted a total of 60 DevRel managers identified at LinkedIn and 18 answered our online survey, giving 85% of confidence level according to the Hamburg's formula (Hamburg, 1980). The applied data collection strategy was 'probability sampling' aiming to eliminate subjectivity and obtain a sample that is both unbiased and representative of the target population. All participants had/have worked with at least one of the following MSECO: Android, iOS, watchOS, Windows Phone, Symbian, and Blackberry. They also work in subsidiaries of those organizations in Brazil, China, USA, Israel, Canada, and Mexico (Table 13). They had an average of 6 (± 3.06) years of professional experience in activities related to developer governance in MSECO.

7.2 Study execution

They were invited to answer a questionnaire with the following question: "*What is the relevance of the following key insights to govern mobile application developers?*" by selecting a number within 0–5 indicating the relevance of a key insight when governing

Table 13 Participants' Profile

ID	Ecosystems	Years	Country
1	Android	4	Brazil
2	Android, iOS, watchOS	2	Canada
3	Windows, Blackberry	4	Brazil
4	Windows	2	México
5	Android, Windows	6	USA
6	iOS, watchOS	2	Brazil
7	Android	2	Brazil
8	Android, iOS, Windows	3	Brazil
9	Android, Windows, Symbian, S40	5	Brazil
10	Android, iOS, Windows	7	Mexico
11	Windows	5	Canada
12	Windows, BlackBerry	3	Brazil
13	Android	4	Brazil
14	Android	9	China
15	Android, iOS, Symbian, Windows	10	USA
16	Android, Windows, Nokia	13	Mexico
17	iOS, watchOS	4	USA
18	Android, iOS, Web	11	Israel

mobile app developers, where: 0 -No relevance, 1 - Low Relevance, 3 - Medium Relevance, and 5 - High Relevance.

7.3 Results' analysis and discussion

Table 14 presents the key insights ordered according to its level of relevance.

We can see that the two most relevant strategies are related to the identification of the community of experts within an MSECO and to the reputation of such community. These two key insights provide an indication that it is more relevant to work by fostering a community of expert developers who can assist in supporting the community. These developers can act as “influencers” in the community. From a community management perspective, it allows keystone to coordinate experts' activities and they can also create and support opportunities for contributions within an MSECO. These key insights show us that it is necessary to allow developer awareness through influencers that are the honeypots of any good communications activity. The influencers help organizations in creating spaces for inclusion and build virtual places where developers can thrive, uncovering the obstacles to community engagement and offering a clear path to developer community. Therefore, they could understand how to get from one phase to the next.

The key insights derived from the analysis of the researchers involved in this work, through the use of peer review. After undergoing analysis of key insights' relevance by DevRel practitioners, we performed an association of the key insights with a set of developer governance strategies previously identified in (Fontao et al., 2017). In the next subsection we describe the association of strategies with key insights.

7.4 Connecting key insights and strategies to govern developers

After verifying the relevance level, we associated the key insights with a set of strategies extracted from a systematic mapping study (Fontao et al., 2017) on developer governance in SECO. The strategies are related to studies that have carried out the evaluation in real SECO scenarios. Table 15 shows the association between key insights and a set of strategies. For each strategy we present the identifiers (Appendix: Table 16) of the

Table 14 Key insights' relevance level

Ranking	Key Insight	Relevance Level
1	The most visualized topics as well as the topics in which developers are most committed to respond can indicate a community of experts who can help to reduce frequent barriers to participation in MSECO.	4 (± 0.8)
2	Badges can help a keystone to manage strategies related to technical resource exploration, active developer in the community, and community control by fostering relationships with top developers in the ecosystem.	3.8 (± 1.1)
3	Questions posted in Stack Overflow next to official MSECO announcement periods can help a keystone to manage strategies to add new MSECO resources (e.g. platforms, SDKs, APIs, programming languages). When such new technologies are released to the market, a keystone should be able to manage them easily.	3.7 (± 1.2)
4	The most commonly used tags in recently added questions may indicate the most frequent barriers faced by developers willing to participate in an MSECO. This scenario can serve as a monitoring strategy to support a keystone in recruiting and educating developers.	3.2 (± 1.5)

Table 15 Key Insight and Strategies

Key Insight and Strategies

The most visualized topics as well as the topics in which developers are most committed to respond in Stack Overflow can indicate a community of experts who can help to reduce frequent barriers to participation in an ecosystem.

- **GD01:** [CD] Meeting the developers' needs to maintain absorptive capacity and avoid discouraging them [S48];
- **GD02:** [CD] Modeling participants, their connections and flows describing the type of product that flows down these connections [S08];
- **GD03:** [OOC] Providing a platform for organizations and individuals to create and provide training [S14];
- **GD04:** [VC] Supporting developers in reusing functionality from others since they could never have built apps independently [S10].

The information about developer reputation can help to manage strategies related to technical resource exploration, active developers in the community, and community control by fostering relationships with top developers in the ecosystem.

- **GD05:** [CD] Analyzing the lifecycle of developer engagement from business and partner management [S52];
- **GD06:** [VC] Developing a system in which the talented developers are chosen and moved ahead based on the basis of their achievement [S45].

Questions posted in Stack Overflow next to official developers' events announcement periods can help an organization to manage strategies to add new ecosystem resources (e.g. platforms, SDKs, APIs, programming languages). When such new technologies are released to the market, a manager should be able to manage them easily.

- **GD07:** [OOC] Offering developers' conferences to allow them to discuss directions of the underlying technological platform [S51];
- **GD08:** [OOC] Allowing free import/export of ideas and knowledge concerning products, processes, and business models [S34].

The most commonly used tags in recently added questions posted in Stack Overflow may indicate the most frequent barriers faced by developers willing to participate in an ecosystem. This scenario can serve as a strategy to support in recruiting and educating developers

- **GD09:** [CD] Analyzing barriers for participation of developers and potential remedies to them [S01];
- **GD10:** [OOC] Establishing a platform for publication and propagation of error reports, performance measures etc. [S26]

papers from which they were extracted from the systematic mapping study (Fontao et al., 2017). This association was driven from the goal of key insight, for example, if it mentioned "reputation", the strategies covering reputational tasks were associated to it. These strategies could help to implement developer governance tasks related to a specific key insight. To do so, we used categories of governance mechanisms described in Section 3: CD (Coordination of Developers), OOC (Organizational Openness and Control), and VC (Value Creation). Based on the Stack Overflow mining study and in the proposal and relevance analysis of key insights and strategies for developer governance extracted from systematic mapping, we propose some actions for keystones that want to offer appropriate governance in MSECO.

GD01 is used when the organizations are willing to assure that the technology continued to meet their respective needs, to maintain absorptive capacity and to avoid discouraging external innovators. For this strategy, it is suggested that the organization use Q&A repositories as a way to monitor existing specialists in the developer community who can act on reducing barriers. We suggest keystone works on recognition mechanisms from the expert community providing more advanced training. Another action may be the insertion into a monitoring mechanism (e.g. dashboard) for the organization's DevRel professionals that allows the visualization of the experts in the developer community.

GD02 is related to understand the governance from software product context and the supply network. A supply network displays all participants, their connections, and flows describing the type of product that flows down these connections. The

organization in this scenario can use such strategy to understand that niches of communities produce and consume contributions, which favors the direction of action in specific niches for the creation of value in the products of the MSECO. For example, if the organization has a community of developers (including experts) who generate knowledge and contributions to a product, this may indicate product acceptance by that developer community and even users. Therefore, investment in product marketing and developer marketing can be driven by keystone.

GD03 comprises a training designing solution to provide a platform for organizations and individuals create an interactive training content to target audiences, as well as to track the effectiveness of related training sessions. This strategy tells keystones that it should not only focus on developer governance in mobile application developers, but also on developers who produce technical knowledge about SDKs, APIs, Tools, IDEs, and any other features of the platform owned by MSECO. This reduces the cost of producing technical material and engages community specialists as it reduces participation barriers. Thus, DevRel professionals will be able to coordinate the creation and maintenance of content by the community by indicating what content the community needs to consume.

GD04 is related to provide opportunistically and pragmatically reuse. Opportunistic reuse serves as extending software with functionality from a third-party software supplier that was not originally intended to be integrated and reused. The pragmatic reuse is related to extending software with functionality from a third-party software supplier that was found without a formal search-and-procurement process and might not have been built with a reuse mindset. Some keystones such as Google already invest in DevRel employees who are participating in Q&A repositories such as the Stack Overflow as a way to check existing code snippets that are built by developers as parts of responses. In addition, members of organizations can enter Stack Overflow to act on the reputation of questions and answers and thereby improving the quality of content for MSECO developers.

GD05 is a guideline focused on the business model and partner management lifecycle in an ecosystem. It includes developers' goals, "enablers" to reach those goals, "effects" describing partners' perceptions of the partnership, and "influencers". The "instruments" can be interpreted as concrete instances of the "enablers". This strategy can be implemented by analyzing the engagement flow of developers within the Stack Overflow, with analysis including primary emotions such as joy, anger and sadness, for example. The developer can still be monitored by the organization from the very first post and his/her responses within the Stack Overflow, his/her problems, what technical resources he/she uses, and whether he/she has changed development strategy to generate some contribution. This also serves to analyze at what level the developer contribution is aligned with the product roadmap for the developer (i.e. APIs, Emulators, IDEs) and the user (i.e. mobile devices with new functionality or hardware capabilities). When we discuss the dashboard for DevRel team monitoring, such strategy supports the identification of several developer niches and also an overview of MSECO status from the developers' point of view.

GD06 suggests an organization to invest in increasing developer reputation and benefits (e.g. future job opportunities), i.e. if the developer invests in the app he/she may invest on the platform too. This scenario is related to the recognition of top developers, that is, developers who produce impact contributions to the MSECO that strengthen

keystone's relationship with the community. This strategy can be implemented by focusing on the developer's marketing activities, such as disclosure of the success story of developers on official MSECO channels, and meetings between DevRel team and top developers. Another suggestion for implementing this strategy is to create and monitor an ambassadorial program (e.g. Google Experts and Microsoft MVP); these programs offer early access to development tools, trainings, discounts at official MSECO events, and prestige of recognition in the community. The use of recognition strategies favors a core DevRel team and allows it to focus on supporting top developers in a more "health" approach while another part of the team may be focused on a "breath" approach to reach out to more members of the community.

GD07 involving a developer conference facilitates decisions about the technological platform. Conferences aim at bringing together technical leaders of all actors in the ecosystem to allow discussion of future directions of the underlying platform and concepts of the ecosystem, including actor-interdependencies. By analyzing Stack Overflow technical issues, we realize that developers are already starting commenting on early versions of the products that will be released at conferences. Therefore, a keystone can use it to direct the speech and even focus on the diffusion of an ad that produces the greater impact for its products.

GD08 suggests keystones to allow free ("open") import and export of ideas and knowledge concerning products, processes and business models that flow between organizations and their environments in order to improve communication between people. Indeed, more openness will provide a larger set of possible business opportunities.

GD09 is a primordial strategy that refers to the identification of barriers for the participation of developers and the mitigation of them through "remedies" as a way to prevent or reduce their effect on the community. For this strategy, we suggest that a DevRel team defines a risk mitigation plan from the possible barriers that can be faced by developers within the MSECO. The risk mitigation plan consists of risks identification, analysis, planning, tracking and communication as well as how to mitigate them.

Finally, **GD10** is about establishing a platform for publication and propagation of acquired knowledge to developers (e.g. error reports, performance measures etc.). This strategy can be implemented by disseminating knowledge-based announcements about problems regarding MSECO technical resources, tools and products. This publication and propagation can also be done by using portals of communication with developers (e.g. Apple Developer, Android Developer, and Microsoft Developer), as well as social networks of DevRel professionals playing in the ecosystem.

7.5 Threats to validity

Below we present the possible threats to validity involved in this study, and how we mitigated it.

Constructo validity: the study is characterized by relevance analysis of the key insights with respect to the current activities required by the developer governance in MSECO. Participants were not involved in other experiments during the survey execution.

Internal validity: In the survey, we proposed to select practitioners (managers) who work in the main MSECOs. Thus, we assumed that they are representative for the population of practitioners involved in developer governance. The questionnaires were reviewed and submitted to a pilot study.

External validity: regarding the survey as mentioned in internal validity, the participants act in the main MSECOs. However, new studies could be performed with more practitioners.

Conclusion validity: it was accomplished through simple demonstration of relevance (or not) of key insights. The comments were obtained directly from forms answered by practitioners without researchers' intervention.

8 Conclusion

App developers use Q&A repositories as a way to solve technical questions that arise throughout the app (and platform) development process. An example of a Q&A repository is the Stack Overflow, with more than 7.5 billion visits⁴ until in 2017 (latest report). In this context, developers, technical resources, apps and other elements have been studied as MSECO. MSECO can be treated as a hybrid ecosystem, since it has a proprietary platform structure, but it is influenced by the use of external repositories controlled by their communities. In this scenario, Stack Overflow holds relevant information about the developer and their participation in MSECO.

In this study, we analyzed the three main MSECO: Android, iOS, and Windows Phone. We mined 1,568,377 technical questions at Stack Overflow aiming to identify what can be understood about MSECOs. We found relevant information involving the most visualized and answered questions, developer engagement, relation between questions and official events, and developer reputation. After analyzing the results obtained to each question, we identified a set of four key insights (or propositions) that can help to understand the involvement of developers in MSECO. In addition, we shared a set of datasets containing data from 2008 to December 2017 that can be used by researchers as a way to study the community of developers in other types of ecosystems. The existing set of information serves both the support of the developer community and the organization itself that can evaluate the effect of adopting SDKs, for example. We concluded that a keystone can use Stack Overflow as an external repository since it is a source of information for the creation and adaptation of ecosystem strategies. As such, data extracted from a Q&A repository can be used as input to support ecosystem's information visualization.

We also investigated how MSECO developers' governance mechanisms can be identified and supported by Q&A repositories. To do so, we performed an evaluation of key insights' relevance by 18 practitioners who work/worked with developer governance in MSECOs. We noted that key insights focused on community reputation and expertise are most relevant for governing a developer ecosystem. Then, it was possible to associate 10 strategies with the four key insights in order to indicate ways for an organization to practice the use of knowledge extracted from Q&A repositories. We know that there is a need for an evaluation of the strategies by both developers and members. There is also a need for a reference model to support both industry and research in the area of developer governance. As future work, we are exploring complex network analysis, fine-grained emotion detection, and MSECO lifecycle through the analysis of questions and answers. It is also important to understand the correlation between data from Q&A repositories and information retrieved from other repositories such as Apps' Store, Social Sites (Facebook and Twitter), Github, and CodePlex.

1 Appendix

Table 16 STUDIES

#	Title
S01	Nagy, D., Yassin, A.M. and Bhattacharjee, A., 2010. Organizational adoption of open source software: barriers and remedies. <i>Communications of the ACM</i> , 53(3), pp.148–151.
S08	Jansen, S., Finkelstein, A. and Brinkkemper, S., 2007. Analyzing the Business of Software: A Modelling Technique for Software Supply Networks. In CAiSE Forum.
S10	Jansen, S., Brinkkemper, S., Hunink, I. and Demir, C., 2008. Pragmatic and opportunistic reuse in innovative start-up companies. <i>IEEE software</i> , 25(6).
S14	Faraz, S.M., Behlim, S.I., Khan, S.M. and Sattar, S.A., 2009, December. Interactive training framework a new approach to eLearning. In <i>Proceedings of the 7th International Conference on Frontiers of Information Technology</i> (p. 79). ACM.
S26	van der Schuur, H., Jansen, S. and Brinkkemper, S., 2011, November. The power of propagation: on the role of software operation knowledge within software ecosystems. In <i>Proceedings of the International Conference on Management of Emergent Digital EcoSystems</i> (pp. 76–84). ACM.
S34	Krogstie, J., 2012. Modeling of digital ecosystems: Challenges and opportunities. <i>Collaborative Networks in the Internet of Services</i> , pp.137–145.
S45	Eckhardt, E., Kaats, E., Jansen, S. and Alves, C., 2014, August. The merits of a meritocracy in open source software ecosystems. In <i>Proceedings of the 2014 European Conference on Software Architecture Workshops</i> (p. 7). ACM.
S48	Teixeira, J. and Lin, T., 2014, May. Collaboration in the open-source arena: the webkit case. In <i>Proceedings of the 52nd ACM conference on Computers and people research</i> (pp. 121–129). ACM.
S51	Knauss, E., Damian, D., Knauss, A. and Borici, A., 2014, August. Openness and requirements: opportunities and tradeoffs in software ecosystems. In <i>Requirements Engineering Conference (RE), 2014 IEEE 22nd International</i> (pp. 213–222). IEEE.
S52	Rickmann, T., Wenzel, S. and Fischbach, K., 2014. Software ecosystem orchestration: the perspective of complementors.

9 Endnotes

¹<https://developer.android.com>

²<https://developer.apple.com>

³<https://www.google.com/intl/km/about/careers/teams/client-facing/dev-rel/>

⁴[https://www.theverge.com/2017/10/9/16446280/](https://www.theverge.com/2017/10/9/16446280/microsoft-finally-admits-windows-phone-is-dead)

[microsoft-finally-admits-windows-phone-is-dead](https://www.theverge.com/2017/10/9/16446280/microsoft-finally-admits-windows-phone-is-dead)

⁵NLTK – Natural Language ToolKit (<http://www.nltk.org>) is a leading platform for building Python programs to work with human language data.

⁶TF-IDF – Term Frequency-Inverse Document Frequency is a numerical statistic that reflects how important a word is to a document in a collection or corpus.

Abbreviations

API: Application programming interface; LDA: Latent dirichlet allocation; MSECO: Mobile software ecosystem; MSR: Mining software repositories; Q&A: Questions and answers; RQ: Research question; SDK: Software development kit; SECO: Software ecosystem; WP: Windows phone

Acknowledgments

We thank FAPEAM, CAPES and CNPq for the financial support. The sixth author also thanks to DPq/PROPG/UNIRIO for partially support this research.

Funding

FAPEAM, CAPES and CNPq sponsored this work.

Availability of data and materials

The source code and the link to download the algorithms can be found at <https://goo.gl/eLYRWN>

Authors' contributions

AF wrote the background, related work and participated in the design of the studies and performed the results' analysis. BA wrote the algorithm of relevant topics detection and participated in results' analysis. IW participated in the

design of the studies and reviewed the analysis of research questions answers. BE participated in survey design and reviewed the results. MQ participated in reviewing the results' analysis. RPS helped to draft the manuscript and participated in the results' analysis of the studies as the PhD co-advisor of AF. ACDN reviewed the design, statistical analysis and results of mining software repositories and survey as the PhD advisor of AF. All the authors contributed to reviewing the paper. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Institute of Computing, Federal University of Amazonas, Manaus, AM, Brazil. ²Federal University of Technology - Paraná, Campo Mourão, PR, Brazil. ³IR, Independent Researcher, Porto, Portugal. ⁴Federal University of Goiás, Goiânia, GO, Brazil. ⁵Federal University of the State of Rio de Janeiro, Rio de Janeiro, RJ, Brazil.

Received: 8 February 2018 Accepted: 10 August 2018

Published online: 22 August 2018

References

- Ahmad A, Feng C, Ge S, Yousif A (2018) A survey on mining stack overflow: question and answering (Q&a) community. *Data Technologies and Applications* 52(2):190–247
- Ahmed H (2008) The road ahead for mining software repositories. In: *Proceedings of the Frontiers of Software Maintenance*, pp 48–57
- Albert, Benno E.; Santos, Rodrigo P.; Werner, Cláudia ML. (2013) Software ecosystems governance to enable it architecture based on software asset management. *Proceedings of the 7th DEST*, p. 55–60
- Alves C, Oliveira J, and Jansen S. "Software Ecosystems Governance - A Systematic Literature Review and Research Agenda," *Proc. 19th Int. Conf. Enterp. Inf. Syst.*, 2017, pp. 215–226
- Axelsson J, Skoglund M (2016) Quality assurance in software ecosystems: a systematic literature mapping and research agenda. *J Syst Softw* 114:69–81 <https://doi.org/10.1016/j.jss.2015.12.020>
- Baars A, Jansen S (2012) A framework for software ecosystem governance. In: *International conference of software business*. Springer, pp 168–180
- Bajaj K, Mesbah A (2016) Mining questions asked by web developers. In: *Proceedings of the International Conference on Mining Software Repositories*, pp 112–121
- Barbosa O, Alves C (2011) A systematic mapping study on software ecosystems. In: *Proceedings of the Third International Workshop on Software Ecosystems*, pp 15–26
- Barua A, Thomas SW, Hassan AE (2014) What are developers talking about? An analysis of topics and trends in stack overflow. *Empir Softw Eng* 19(3):619–654
- Basili, V., Heidrich, J., Lindvall, M., Munch, J., Regardie, M. and Trendowicz, A., 2007. GQM^+ strategies—aligning business strategies with software measurement. In *empirical software engineering and measurement, 2007. ESEM 2007. First international symposium on* (pp. 488–490). IEEE
- Bhat V (2014) Min (e) d your tags: analysis of question response time in stack overflow. In: *Proceedings of the International Conference on Advances in Social Network Analysis and Mining*, pp 328–335
- Bosch J (2009) From software product lines to software ecosystems. In: *Proceedings of the International Software Product Line Conference*, pp 111–119
- Casalnuovo C, Vasilescu B, Devanbu P, Filkov V (2015) *Proceedings of the Joint Meeting on Foundations of Software Engineering*. In: *Developer onboarding in GitHub: the role of prior social links and language experience*, pp 817–828
- de Souza C, Filho F, Miranda M, Ferreira R, Treude C, Singer L (2016) The social side of software platform ecosystems. In: *Proceedings of the International Conference on Human Factors in Computing Systems*, pp 3204–3214
- Eckhardt E, Kaats E, Jansen S, Alves C (2014) The merits of a meritocracy in open source software ecosystems. In: *Proceedings of the European Conference on Software Architecture*, p 7
- Farias M, Novais R, Colaço M, Carvalho L, Mendonça M, Spinola R (2016) A systematic mapping study on mining software repositories. In: *Proceedings of the ACM/SIGAPP Symposium on Applied Computing*, pp 1472–1479
- Foerderer J, Kude T, Schuetz SW, e Heinzl A (2018) Knowledge boundaries in enterprise software platform development: antecedents and consequences for platform governance. *Information Systems Journal* 28(1):1–26
- Fontao A, Estácio B, Wiese I, Santos R, Dias-Neto A (2017) Governing developers in software ecosystems. Technical Report. Available at: <https://goo.gl/uA5DJu>
- Fontão A, Santos R, Filho JF, Dias-Neto AC (2016) MSECO-DEV: application development process in mobile software ecosystems. In: *Proceedings of the International Conference on Software Engineering and Knowledge Engineering*, pp 317–322
- Fontão A, Santos RP, Dias-Neto AC (2015) *Proceedings of the Annual International Computers, Software & Applications Conference*. In: *Mobile software ecosystem (MSECO) (ed) a systematic mapping study*, pp 653–658
- Genc-Nayebi N, Abran A (2016) A systematic literature review: opinion mining studies from mobile app store user reviews. *J Syst Softw* 125:207–2019
- German D, Adams B, Hassan AE (2013) The evolution of the R software ecosystem. In: *Proceedings of the European Conference on Software Maintenance and Reengineering*, pp 243–252
- Hamburg M. "Basic statistics: A modern approach," *J. R. Stat. Soc.*, vol. 143, 1980, no. 1
- Koch S, Kerschbaum M (2014) Joining a smartphone ecosystem: Application developers' motivations and decision criteria. *Inf Softw Technol* 56(11):1423–1435

- Krestel R, Fankhauser P, Nejdl W (2009) Latent dirichlet allocation for tag recommendation. In: Proceedings of the third ACM conference on Recommender systems, pp 61–68
- Larson, Ray R. "Introduction to information retrieval." *Journal of the American Society for Information Science and Technology* 61, no. 4 (2010): 852–853
- Lin F, Ye W (2009) Operating system battle in the ecosystem of smartphone industry. In: Proceedings of the International Symposium on Information Engineering and Electronic Commerce, pp 617–621
- Loper, E. and Bird, S., 2002. NLTK: the natural language toolkit. In proceedings of the ACL-02 workshop on effective tools and methodologies for teaching natural language processing and computational linguistics-volume 1 (pp. 63-70). Association for Computational Linguistics
- Mäenpää, H., Munezero, M., Fagerholm, F., Mikkonen, T. 2017. The many hats and the broken binoculars, in: proceedings of the 13th international symposium on open collaboration - OpenSym '17. Pp. 1–9. <https://doi.org/10.1145/3125433.3125474>
- Manikas K (2016) Revisiting software ecosystems research: a longitudinal literature study. *J Syst Softw* 117:84–103
- Manikas K, Hansen KM (2013a) Software ecosystems – A systematic literature review. *J Syst Softw* 86(5):1294–1306
- Manikas K, Wnuk K, Shollo A (2015) Defining decision making strategies in software ecosystem governance. University of Copenhagen, Department of Computer Science
- Manikas K, Hansen KM (2013b) Reviewing the health of software ecosystems—a conceptual framework proposal. In: Proceedings of the 5th International Workshop on Software Ecosystems (IWSECO), pp 33–44
- O'Mahony S (2007) *J Manage Governance* 11:139 <https://doi.org/10.1007/s10997-007-9024-7>
- Rosen C, Shihab E (2016) What are mobile developers asking about? A large scale study using stack overflow. *Empir Softw Eng* 21(3):1192–1223
- Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 20:53–65
- Sadi MH, Dai J, Yu E (2015) Designing software ecosystems: how to develop sustainable collaborations? *Proceeding of the CAiSE 2015(p)*:161–173
- Santos R, and Werner C (2012) ReuseECOS: an approach to support global software development through software ecosystems. In Proceedings of the IEEE International Conference on Global Software Engineering Workshops, 60–65
- Schreieck, M., Wiesche, M., Krmar, H. (2016). Design and governance of platform ecosystems – key concepts and issues for future research. *Ecis* 1–20
- Shah C, Kitzie V, and Choi E (2014) Questioning the question - addressing the answerability of questions in community question-answering. In Proceedings of the Annual Hawaii International Conference on System Sciences, 1386–1395
- Shull F, Singer J, and Sjøberg DIK (2008) Guide to advanced empirical software engineering
- Song J, Baker J, Wang Y, Choi HY, Bhattacharjee A (2018) Platform adoption by mobile application developers: a multimethodological approach. *Decis Support Syst* 107:26–39 <https://doi.org/10.1016/j.dss.2017.12.013>
- Valença G, Alves C (2017) A theory of power in emerging software ecosystems formed by small-to-medium enterprises. *J Syst Softw* 134:76–104 <https://doi.org/10.1016/j.jss.2017.08.044>
- Wareham J et al (2014) Technology ecosystem governance. *Organ Sci* 25(4):1195–1215
- Zagalsky A, Teshima CG, German DM, Storey M, Poo-caamaño G (2016) How the R community creates and curates knowledge: a comparative study of stack overflow and mailing lists. In: Proceedings of the International Conference on Mining Software Repositories, pp 441–451

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
