Squirrel: A Code Snippet Repository

Aziz Nanthaamornphong, Saknarong Pomwong, Kwanrudee Klebkaew, and Napawan Jindamanee
Faculty of Technology and Environment, Prince of Songkla University, Phuket Campus, Thailand.
aziz.n@phuket.psu.ac.th

Abstract—**Teaching programming courses in an academic environment has challenges, particularly for undergraduate students. These challenges can also be found in the software industry, where novice developers still need to obtain coding training. In practice, lecturers or trainers always assign exercises to students with the goal of improving their coding skills. Based on teaching experience, we have found that students may need to rely on software development kits to code an application faster, such as reusing lines of code already developed (code snippets) to solve a particular problem. Additionally, to reduce the software development time, novice developers tend to use crowdsourcing and social media sources of example code that can be used to solve programming problems. Utilizing these code snippets not only saves time for students or novice developers but also helps them to learn. Therefore, we proposed a system, called Squirrel, that is capable of offline and online code snippet storage and searching. Additionally, for online functionalities, this application supports searching for code snippets that exist in question-and-answer websites, such as Stack Overflow. We believe that this system will help increase the effectiveness of software development, especially for computer-science students and novice developers.**

Index Terms—**Software Development; Software Engineering; Software Engineering Education; Stack Overflow.**

I. INTRODUCTION

Currently, developers benefit from example code in online repositories to solve their programming problems by iteratively searching for, adjusting, and merging examples [1-4]. They can reuse existing code examples or adapt the code to the assigned problems instead of starting from scratch [5], thereby accelerating development and increasing accuracy, since the existing code is likely to be demonstrated to work [6]. Consequently, if developers have the proper code, they can solve problems more efficiently [7]. The academic environment is no different: undergraduate students or novice developers are unable to memorize every detail of a programming language's syntax. Therefore, they have to look for code snippets on the Internet, especially from popular community sites such as Stack Overflow, which has a large amount of high-quality code [8], to be used as a guideline for solving assigned tasks; examples include database connection and array collection. Based on the recent results of the 2017 Stack Overflow Developer Survey, most developers think that Stack Overflow provides helpful answers and code examples [9]. For example, a previous study reported that 20% of jQuery-related questions have a code snippet embedded in a Stack Overflow accepted answer [10].

However, this technique requires online searching, which requires an Internet connection. In some situations, developers must work in an environment where having an Internet connection at all times is not possible, and hence they are unable to search online. In addition to the Internet connection problem, searching for code online will produce results from many sources; it thus takes time to choose the right solution for a given problem.

To mitigate these problems, we propose Squirrel, a code snippet repository that integrates the crowd knowledge of question-and-answer (Q&A) websites. Squirrel helps users to:

i. Accelerate software development by making use of code snippets that have been proven to work
ii. Search for programming solutions by connecting with popular Q&A websites, such as Stack Overflow

Using these code snippets will help novice developers and undergraduate students save time and aid in their learning. The system allows users to search for and store code snippets. Additionally, the proposed system provides an opportunity for users to share their knowledge about software development. This system supports offline and online access to help developers easily select code snippets or solutions to problems. While working in online mode, users can also search for solutions from Stack Overflow and other Q&A websites included in the Stack Exchange [11]. Although existing studies indicate that Q&A websites do not provide high-level technical answers [12-14], these websites contribute to the body of knowledge in software development [15]. Additionally, extracting the knowledge from crowdsourced sites and social media networks is an essential practice in software development communities [16].

We expect that the proposed system would have the following benefits:

i. Novice developers, especially undergraduate students, could engage in more effective software development.
ii. The code snippets might improve the quality of source code because this code has been written by experienced developers and proven to work.

The remainder of this paper is organized as follows. Section II provides an overview of the relevant background. In Section III, the implementation of Squirrel is presented. Section IV discusses the results of the tool's evaluation. Finally, conclusions and future work are discussed in Section V.

II. RELATED WORK AND BACKGROUND

Some existing tools [17-18], which help programmers work on software development inspired us to develop Squirrel. Although different code search tools were proposed [19-21], all of them rely on the Internet capability. Sahavechaphan and Claypool developed XSnippet [22], which enables to suggest relevant code snippets for the programming task. However, the suggested code snippets are selected from a limited sample repository. Nevertheless, unlike Squirrel, none of above approaches provide an approach that integrates the Q&A websites within their tools. The developed system was based on a few primary

technologies, including web services and the Stack Exchange API. Details of each technology are presented below.

A. Web Services

Web services are a technology for data exchange between systems that may run on a computer system in a network [23]. Web services enable systems developed on different platforms or programming languages to exchange data between each other based on the World Wide Web Consortium (W3C) standard [24], which defines a web service in the form of Uniform Resource Identifier (URI). Generally, the exchanged content in web services is an eXtensible Markup Language (XML)-based file.

To develop this system, we utilized web services in three areas:
  i.   updating code snippets
  ii. searching for code snippets
  iii. searching Q & A websites

B. Stack Exchange

Stack Overflow is a huge online community for posting and answering questions (currently, it has more than 6.9 million users, 14 million questions, and 21 million answers) [25]. It uses social mechanisms to facilitate knowledge exchange among users. A user can ask a question about almost any programming-related topic and receive a response within a median time of 10 minutes [14]. Only the best answer is selected to be "the accepted answer" (represented by a green check mark).

In fact, Stack Overflow is connected to a large portal, called Stack Exchange, that includes multiple Q & A websites, such as Server Fault [26], Ask Ubuntu [27], Database Administrators [28], Android Enthusiasts [29], and Game Development [30]. Those websites and community networks aforementioned have different purposes, as described below.

  i.   Stack Overflow - it is a Q&A website that allows the user to discuss the programming problems, including bug issues, frameworks, and tools.
  ii. Server Fault - it is a Q&A website focusing on computer hardware issues, such as server computers and networking. This website is suitable for computer administrators.
  iii. Ask Ubuntu - it is a Q&A website for Ubuntu users.
  iv. Database Administrator - it is a Q&A website for database professionals who wish to enhance their database skills.
  v.   Android Enthusiasts - it is a Q&A website for persons who are interested in a particular Android operating system.
  vi. Game Development - questions and answers about software development are discussed on this website.

Stack Exchange has APIs to enable developers to search data in its databases by calling web services; these are termed Stack Exchange APIs. The search pattern is based on a similar service request category, such as the following:

  i. Answer: This is a collection of commands to search for answers from posts.
  ii. Post: This is a collection of commands to search for posts.

Figure 1 shows a sample URL for calling Stack Exchange APIs to retrieve posts with the keyword "array" on Android Enthusiasts by sorting vote scores in descending order. The result is in the form of JavaScript Object Notation (JSON), which is an object-oriented JavaScript, thereby making it easier to manage and access data.

```
https://api.stackexchange.com/2.2/search/advanced
?order=desc&sort=votes&title=array&site=android
```

Figure 1: Example URL of Stack Exchange API

III. UNDER THE HOOD OF SQUIRREL

Squirrel consists of 3 parts: 1) web application, 2) web services, and 3) desktop application. Figure 2 shows the overall system architecture. Details of each part will be described in the following sections.
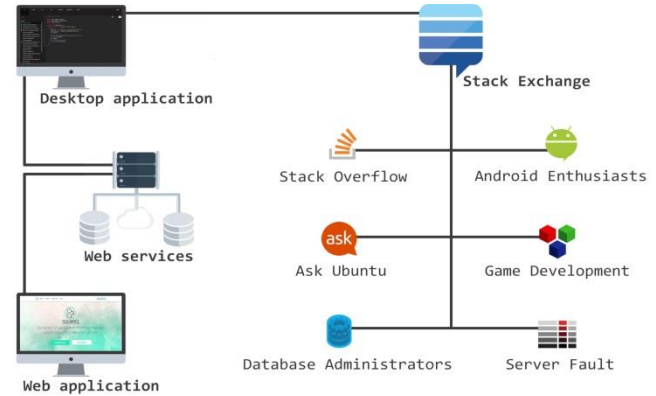

Figure 2: System Architecture

A. Web Application

This web application aims to create a community where developers can share not only code snippets but also stories (blogs) related to software development. There are two types of users in the system: one is called a normal user, and another is a member. A normal user has viewing rights only, whereas a member has the rights to share code snippets and software development stories.

Additionally, the web application has a back office for administrators (shown in Figure 3). The back office allows an administrator to manage the website in the following areas.

  i.   Code snippet management that supports code snippets written in C#, Java, Python, and VB.NET. This serves new code snippets that can be downloaded to a desktop application
  ii. User management to change website access rights
  iii. Sharing management to control inappropriate user content or annoyance of another user
  iv. Statistics regarding web application usage in a graphical form.

B. Web Services

Behind the developed system, the web services were developed to communicate with the web application, database, and desktop application. One of the main functions of the web services is to update data at the desktop application due to changes to code snippets in the system. To do so, the web services compare the last updated date of the data in the desktop application to the date in the database. If the data in the desktop application are up-to-date, no data will be retrieved from the database, but if the data in the desktop application are outdated, the web services will transmit the up-to-date data to the client. Another function of the web services is to search code snippet titles, which are shared by

users on the web application and displayed on the desktop application.

However, for the first-time installation of the desktop application, there is no need to obtain data from the web services, since data in the program is most up-to-date. In our system, the format of data is JSON, as shown in Figure 4. In the update process, JSON is extracted to following data fields:

- i. Id – the unique number of the data entry
- ii. Title – the title of the code snippet
- iii. Programming_language – the programming language of the code snippet
- iv. User_id – the user who created the code snippet
- v. Created_at – the time and date that the user created the code snippet
- vi. Update_at – the last updated time and date of the code snippet

Figure 3: Administrator's Website

```
{
    snippets:[
        {
            id:1,
            title:"calculate the GCD and LCM of two given numbers",
            programming_language:"java",
            source_code:"//Java code here",
            language_id:1,
            user_id:1,
            created_at:"2017-03-09 17:45:15",
            updated_at:"2017-03-09 17:45:15"
        },
        {
            id:2,
            title:"Find the Unique Factorization of a Given Number",
            programming_language:"java",
            source_code:"//Java code here",
            language_id:1,
            user_id:1,
            created_at:"2017-03-09 17:45:15",
            updated_at:"2017-03-09 17:45:15"
        }
    ]
}
```
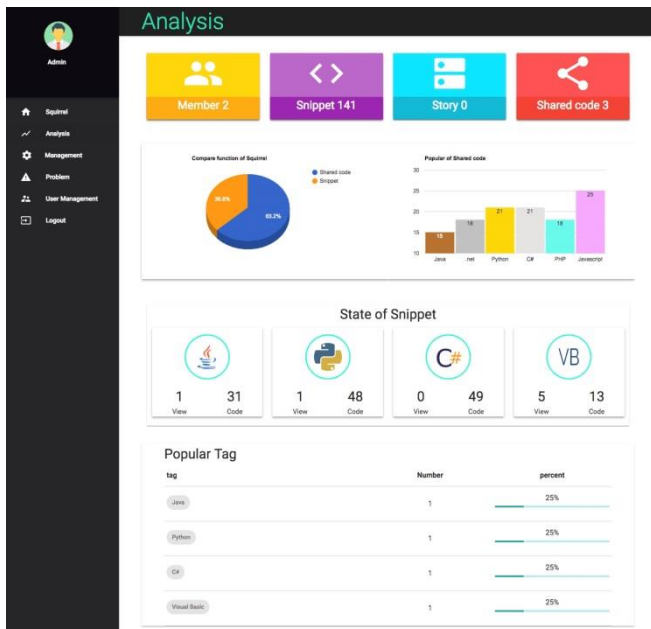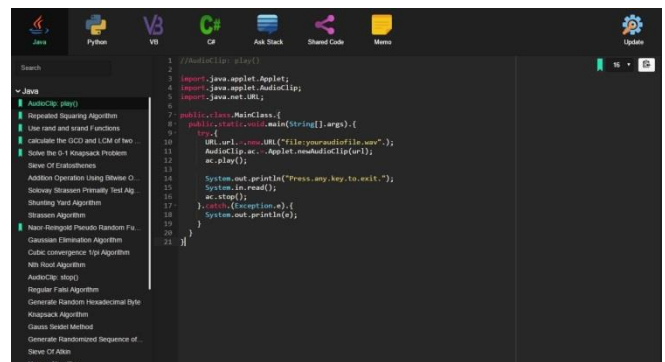
Figure 4: Example of JSON

Figure 5: Desktop Application

IV. SQUIRREL

In this section, we describe how Squirrel is used and evaluate its user satisfaction.

A. User Scenarios

For the web application, users are categorized into three groups: normal users, members, and administrators. A normal user is able to view code snippets in the Snippet section. Those code snippets are added by administrators, in addition to the samples in the Shared Code section, which are added by members. The code snippets added by administrators are obtained from trusted sources, such as algorithm books, programming learning books, and research articles.

Furthermore, normal users can access software development stores in the Story section by clicking on a title of an interesting topic. However, normal users are unable to comment, share code snippets or share software development stories with other users.

Normal users can apply for membership of the website to obtain the right to share code snippets and stories, in addition to rights to comment, follow, and click on the Useful link in Shared Code and Story sections posted by another member.

Users with member status must supply a correct username and password before accessing the system. A member can share code snippets in the Shared Code section by selecting 'Share your code' under the Shared Code menu and then completely filling out details, including a title, description,

C. Desktop Application

This part is a standalone application, which is independent of a database system. The program stores data in a file format, and that file keeps the JSON objects described in the early section. For offline functionalities, the application supports code snippets written in C#, Java, Python, and VB.NET and bookmarking those code snippets (as shown in Figure 5).

For online functionalities, there are several useful features for users. First, the Ask Stack function is used to search data about software development by calling Q&A websites; it includes Stack Overflow, Ask Ubuntu, Database Administrators, Android Enthusiasts, Game Development, and Server Fault.

Additionally, the Shared Code function can obtain code snippets posted by users on the web application and record them in the desktop application storage for offline viewing.

Another function is Memo, which can display bookmarked titles of code snippets for four programming languages; C#, Java, Python, and VB.NET, in addition to titles downloaded from the website, and has the ability to set note reminders.

programming language and source code, and defining multiple tags as desired. A member owning posts can edit or delete the posts. There is also the Story menu for the member to share stories about software development.

The member can view code snippets and stories shared by other members. Additionally, the member can take a look at his or her activities from the Profile page, which displays the number of followers, the number of followings, and a timeline that illustrates what code snippets and stories have been shared.

An administrator has to supply a correct username and password before accessing the system. The administrator can view a summary of activities on the website by selecting the Analysis menu, and the system will display graphical information showing statistics of frequently used menus, programming languages whose codes are shared most, a number of code snippets, and users for each programming language.

An administrator can add, edit or delete code snippets in the Snippet section by clicking on the Manage Code menu. This menu will list all titles of code snippets of all programming languages, and the administrator is able to group the list and to search for a title in the page.

The Problem menu is used to view problem reports submitted to administrators, such as annoyance, improper word usage, phishing, and spam. Resolutions of each problem are up to administrator's judgment. The User Management menu lists all users and administrators with clear roles. The single user with the super administration role can add or revoke administration rights of users in the system.

The desktop application based on Node.js has GUI and menus, as shown in Figure 5. The menus include selection of code snippets from different programming languages (C#, Java, Python, and VB.NET), searching Stack Exchange, searching Shared Code on the Squirrel website, bookmarking interesting code snippets (namely, Memo), and updating the code snippets. For searching code snippets, suppose that a user wants to find Java samples; the user has to choose Java as the programming language and fill in the search keywords. Then, the system performs a search based on the given keywords and displays a list of code snippet titles. From there, the user can choose a title to see code snippets as desired and bookmark the interesting code snippets by clicking on an icon in front of the title. This bookmarked title will then be saved under the Memo menu. In case the users would like to update the code snippets, they can simply press the Update button; then, the desktop application will call the web services to update its stored data.

In terms of searching code snippets or questions about software development from Stack Exchange, a user can select the Ask Stack menu and then specify search keywords and websites. Subsequently, the system will list titles matched with the search, and when the user clicks on a title as desired, the system will render a website including the title (shown in Figure 6).

For searching code snippets from Shared Code of Squirrel website, a user can select the Shared Code menu, supply search keywords, and then click the Search button. The system will show a list of titles, and if the user would like to see details of code snippets, he or she can click on a particular title, and the system will then display the results. Lastly, if the user prefers to study the samples, he or she can press the Save button for later viewing from the Memo menu.

Figure 6: Searching in Stack Overflow

B. Evaluation

We empirically evaluated users' satisfaction with the proposed system. We asked the participants, including 40 undergraduate students, to modify a Tetris game based on given tasks (e.g., write a new algorithm to calculate the score or add new some features). We chose these students as the participants because all of them were studying in the software development class. Thus, we believed that the code snippet tool would increase students' productivity. The Tetris game has been implemented as two versions, written in Java and C#. The participants could choose either the Java or C# version. The participants developed the program on the provided computers, which had Squirrel installed on them. Each participant had to work in both the offline and online modes, and we limited the time for each mode to three hours. After the development sessions, we conducted a survey through an online questionnaire. The questionnaire consisted of 5 Likert-scale questions addressing the following topics:

i. The satisfaction with the results in the offline mode
ii. The satisfaction with the results in the online mode
iii. The ease of use of the software
iv. The speed of the search tool in the offline mode

Finally, we received the completed responses from all participants. Note that all responses were anonymous. Table 1 presents the results of the survey. Overall, the participants were satisfied with the software. However, they provided the following feedback:

i. The software should allow the users to set their preferences (e.g., background color and font color)
ii. The code should have better comments
iii. The software should allow users to share code on Facebook

Table 1
Results of users' satisfaction

Question	Strong Agree (%)	Agree (%)	Neither Agree nor Disagree (%)	Disagree (%)	Strong Disagree (%)
In the offline mode, the software provides the satisfied results	37.5	42.5	20	0	0
In the online mode, the software provides the satisfied results	40	47.5	12.5	0	0
The software is easy to use	75	12.5	12.5	0	0
You can quickly find the results in the offline mode	32.5	62.5	5	0	0

V. Conclusion and Future Work

In this study, we proposed a system comprised of a desktop application, web application, and web services with the aim of helping novice software developers by providing them with the ability to reuse developed codes to solve particular problems to increase convenience, save time and improve work accuracy. The demonstration VDO of our application is available at http://research.te.psu.ac.th/aziz/ codesnippet.htm. We hope that this system will help less-experienced software developers and students who are interested in software development increase their confidence in starting software development.

In the future, we plan to enhance the system and to add new features. Moreover, we plan to distribute the system in the form of open source software to anyone interested without any fee.

References

[1] M. Umarji, S. E. Sim, and C. Lopes, "Archetypal internet-scale source code searching," in *Open Source Development, Communities and Quality: IFIP 20th World Computer Congress, Working Group 2.3 on Open Source Software*, September 7-10, 2008, Milano, Italy, B. Russo, E. Damiani, S. Hissam, B. Lundell, and G. Succi, Eds. Boston, MA: Springer US, 2008, pp. 257–263.

[2] R. E. Gallardo-Valencia and S. Elliott Sim, "Internet-scale code search," in *Proceedings of the 2009 ICSE Workshop on Search-Driven Development-Users, Infrastructure, Tools and Evaluation, ser. SUITE '09*, 2009, pp. 49–52.

[3] K. Philip, M. Umarji, M. Agarwala, S. E. Sim, R. Gallardo-Valencia, C. V. Lopes, and S. Ratanotayanon, "Software reuse through methodical component reuse and amethodical snippet remixing," in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, ser. CSCW '12*, 2012, pp. 1361–1370.

[4] J. Brandt, P. J. Guo, J. Lewenstein, M. Dontcheva, and S. R. Klemmer, "Writing code to prototype, ideate, and discover," *IEEE Software*, vol. 26, no. 5, pp. 18–24, 2009.

[5] L. P. Deutsch, "Software reusability," T. J. Bigger staff and A. J. Perlis, Eds., 1989, ch. Design Reuse and Frameworks in the Smalltalk-80 System, pp. 57–71.

[6] J. Sillito, F. Maurer, S. M. Nasehi, and C. Burns, "What makes a good code example? : a study of programming Q&A in StackOverflow," in *Proceedings of the 2012 IEEE International Conference on Software Maintenance (ICSM), ser. ICSM '12*, 2012, pp. 25–34.

[7] A. Heydarnoori, K. Czarnecki, W. Binder, and T. T. Bartolomei, "Two studies of framework-usage templates extracted from dynamic traces," *IEEE Transactions on Software Engineering*, vol. 38, no. 6, pp. 1464–1487, 2012.

[8] S. Subramanian and R. Holmes, "Making sense of online code snippets," in *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR), May 2013*, pp. 85–88.

[9] "StackOverflow Developer Survey Results 2017." [Online]. Available: http: //stackoverflow.com/insights/survey/2017/. [Accessed: 2017-03-30].

[10] C. Parnin and C. Treude, "Measuring API documentation on the web," in *Proceedings of the 2nd International Workshop on Web 2.0 for Software Engineering, ser. Web2SE '11*, 2011, pp. 25–30.

[11] S. Exchange, "Stack Exchange," http://stackexchange.com, accessed: 2017-03-30.

[12] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and Yahoo answers: everyone knows something," in *Proceedings of the 17th International Conference on World Wide Web, ser. WWW '08*, 2008, pp. 665–674.

[13] K. K. Nam, M. S. Ackerman, and L. A. Adamic, "Questions in, knowledge in?: A study of Naver's question answering community," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ser. CHI '09*, 2009, pp. 779–788.

[14] L. Mamykina, B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann, "Design lessons from the fastest Q&A site in the west," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ser. CHI '11*, 2011, pp. 2857–2866.

[15] C. Treude, O. Barzilay, and M.-A. Storey, "How do programmers ask and answer questions on the web? (nier track)," in *Proceedings of the 33rd International Conference on Software Engineering*, 2011, pp. 804– 807.

[16] M.-A. Storey, L. Singer, B. Cleary, F. Figueira Filho, and A. Zagalsky, "The (r) evolution of social media in software engineering," in *Proceedings of the on Future of Software Engineering, ser. FOSE 2014*, 2014, pp. 100–116.

[17] E. M. Wilcox, J. W. Atwood, M. M. Burnett, J. J. Cadiz, and C. R. Cook, "Does continuous visual feedback aid debugging in direct-manipulation programming systems?" in *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, ser. CHI '97*, 1997, pp. 258–265.

[18] S. Chatterjee, S. Juvekar, and K. Sen, "Sniff: A search engine for java using free-form queries," in *Proceedings of the 12th International Conference on Fundamental Approaches to Software Engineering: Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2009, ser. FASE '09*, 2009, pp. 385–400.

[19] S. P. Reiss, "Semantics-based code search," in *Proceedings of the 31st International Conference on Software Engineering, ser. ICSE '09*, 2009, pp. 243–253.

[20] C. McMillan, M. Grechanik, D. Poshyvanyk, Q. Xie, and C. Fu, "Portfolio: Finding relevant functions and their usage," in *Proceedings of the 33rd International Conference on Software Engineering, ser. ICSE '11*, 2011, pp. 111–120.

[21] J. Galenson, P. Reames, R. Bodik, B. Hartmann, and K. Sen, "Codehint: dynamic and interactive synthesis of code snippets," in *Proceedings of the 36th International Conference on Software Engineering, ser. ICSE 2014*, 2014, pp. 653–663.

[22] N. Sahavechaphan and K. Claypool, "Xsnippet: Mining for sample code," *SIGPLAN Not.*, vol. 41, no. 10, pp. 413–430, Oct. 2006.

[23] "Web Services Architecture." [Online]. Available: https://www.w3.org/TR/ws- arch/. [Accessed: 12-Jun-2017].

[24] "Web Services Description Requirements." [Online]. Available: https://www.w3.org/TR/WS-DESC-REQS/. [Accessed: 12-Jun-2017].

[25] "StackOverflow." [Online]. Available: http://stackoverflow.com/tour. [Accessed: 12-Jun-2017].

[26] "ServerFault." [Online]. Available: http://www.serverfault.com. [Accessed: 12-Jun-2017].

[27] "Ask Ubuntu." [Online]. Available: http://askubuntu.com. [Accessed: 12-Jun-2017].

[28] "Database Administrators." [Online]. Available: http://dba. stackexchange.com/. [Accessed: 12-Jun-2017].

[29] "Android Enthusiasts." [Online]. Available: http://android.stackexchange. com/. [Accessed: 12-Jun-2017].

[30] "Game Development." [Online]. Available: http://gamedev.stackexchange. com/. [Accessed: 12-Jun-2017].