

An Empirical Study of Q&A Websites for Game Developers

Arthur Kamienski · Cor-Paul Bezemer

Received: date / Accepted: date

Abstract The game development industry is growing, and training new developers in game development-specific abilities is essential to satisfying its need for skilled game developers. These developers require effective learning resources to acquire the information they need and improve their game development skills. Question and Answer (Q&A) websites stand out as some of the most used online learning resources in software development. Many studies have investigated how Q&A websites help software developers become more experienced. However, no studies have explored Q&A websites aimed at game development, and there is little information about how game developers use and interact with these websites. In this paper, we study four Q&A communities by analyzing game development data we collected from their websites and the 347 responses received on a survey we ran with game developers. We observe that the communities have declined over the past few years and identify factors that correlate to these changes. Using a Latent Dirichlet Allocation (LDA) model, we characterize the topics discussed in the communities. We also analyze how topics differ across communities and identify the most discussed topics. Furthermore, we find that survey respondents have a mostly negative view of the communities and tended to stop using the websites once they became more experienced. Finally, we provide recommendations on where game developers should post their questions, which can help mitigate the websites' declines and improve their effectiveness.

Keywords Q&A communities · Game development

Arthur Kamienski · Cor-Paul Bezemer
Analytics of Software, Games and Repository Data (ASGAARD) Lab
University of Alberta
Edmonton, AB, Canada
E-mail: {kamiensk,bezemer}@ualberta.ca

1 Introduction

The game development industry is growing [46] and hungers for experienced developers that can produce high-quality games [40]. This high demand for skilled game developers needs to be satisfied by the training of new developers in game development-specific abilities, and those seeking to develop games need ways to acquire the knowledge they need to hone their skills. Moreover, amateur and professional game developers alike need to stay updated on the most recent practices and technologies.

Developers in other fields of the software development industry face similar challenges and have shown a preference for question and answer (Q&A) websites (mostly Stack Overflow¹) when it comes to acquiring new knowledge and seeking solutions to the problems they face [51]. Research has shown that these Q&A websites provide a fertile environment for knowledge sharing [8], and a large number of studies analyzed how these websites can help their communities by providing information and assistance [4]. Other studies have found similar results for general purpose [3], social [35], and health Q&A websites [41]. Yet, no research analyzed if and how Q&A websites related to game development help game developers.

Several Q&A websites specialize in game development, amassing hundreds of thousands of posts and offering a rich set of valuable data that could help to understand the game development environment. Studying these websites can provide us with a deeper understanding of the social and technical aspects of game development.

In this paper, we explore the data that we collected from four Q&A websites, three of which target game developers (Unity Answers², the Unreal Engine 4 (UE4) AnswerHub³, and the Game Development Stack Exchange⁴) and Stack Overflow, which targets all types of software developers. We analyze their content and activity to understand how game developers use them and what topics are discussed in them. We also use the 347 game developers' responses we received through a survey about two of these websites to analyze their users' behaviour and opinions. Specifically, we answer the following research questions (RQs):

RQ1. How did the studied game development Q&A communities evolve in terms of user participation?

We analyze the communities' activity and effectiveness over time to understand their ability of fostering an active user base. We also explore factors that are correlated to the changes we observed in the communities. We find that the communities have been in decline, with the number of posts and active users per month, the percentages of answered and resolved questions, and their answer effectiveness, measured as the percentage of answers marked as the accepted solution to their questions, decreasing over time. We also find

¹ <https://stackoverflow.com/>, accessed February 3, 2021.

² <https://answers.unity.com/>, accessed February 3, 2021.

³ <https://answers.unrealengine.com/>, accessed February 3, 2021.

⁴ <https://gamedev.stackexchange.com/>, accessed February 3, 2021.

that growth positively correlates to the release of new products, and negatively correlates to the percentage of answered and resolved questions. The most experienced users have a higher effectiveness and answered and solved a large number of questions, but the overall experience of the community decreased over time.

RQ2. What topics are most frequently discussed by game developers on the studied Q&A websites?

Game development requires specific software development knowledge and a diversified set of skills [45, 58]. However, there is still a lack of understanding of what those skills are, and how prominent they are to the game development community. With this in mind, this question explores the topics discussed by the Q&A websites in their questions, answers, and comments. We find that most of the topics we identified are specific to game development, covering several subjects unique to it. Nevertheless, the most discussed topic in Unity Answers and the UE4 AnswerHub was *Bug reports*, while Stack Overflow and the Game Development Stack Exchange had many posts about object-oriented programming.

RQ3. What are the characteristics of posts from each topic?

Posts from game development Q&A websites about distinct topics may have different characteristics, be that in the way users write or interact with them. In this question, we explore if and how posts from the topics we identified differ in terms of specific aspects, such as the length of their text and the presence of code in them. We also compare posts belonging to a same topic across the studied Q&A websites to identify if there is any distinction in the way the communities discuss them. We show that topics were discussed with varying levels of abstraction and complexity, differing in terms of the presence of code and the length of their posts. Furthermore, we found that Unity Answers and Stack Overflow discussed code more frequently than the UE4 AnswerHub and the Game Development Stack Exchange.

RQ4. How do game developers perceive the studied communities?

We conducted a survey among game developers to provide a better view of how they regard Unity Answers and the UE4 AnswerHub and to complement our previous analyses. We chose to only ask questions about those two websites as their communities focus on their own game engine. We find that most respondents prefer other learning resources, such as online guides and tutorials, despite having accessed the Q&A websites in the past. Most respondents also do not actively contribute to the communities and even show disapproval for them. Moreover, respondents accessed the websites less frequently than they used to.

By answering these questions and bringing new knowledge about the studied game development Q&A communities, we shed light on how their users interact and discuss game development technologies. We also assess the overall health of the communities, identify factors that relate to it, and characterize the topics in which they are the best and the worst at discussing. Our findings help game developers in understanding the characteristics of each of the studied Q&A websites, and we provide recommendations that can help them in

choosing on which community to post their questions. By directing questions to communities that can handle them more effectively, we can help mitigate the communities’ decline and increase their overall effectiveness.

Paper outline: In Section 2 we provide background information on Q&A websites and game engines, and discuss prior work conducted on these topics. Section 3 explains our methodology and Section 4 describes the motivation, approach, and findings for each of our RQs. We discuss the implications of our findings in Section 5, the threats to validity in Section 6, and conclude the paper in Section 7.

2 Background and Related work

In this section, we provide background information on Q&A communities, Q&A websites for game development, and topic modeling of Q&A websites. We also provide an overview of other work related to this paper and to these concepts.

2.1 Q&A communities

Question and Answer (Q&A) websites allow registered users to ask for help from peers by posing questions about the knowledge they seek and the problems they are facing. Users can interact either by asking, answering, or commenting on a question. Some Q&A websites, such as Yahoo! Answers⁵ and Quora⁶, cover a variety of subjects, while others specialize in specific areas like programming, cooking, photography, and academic research. These websites host large communities of users that share the same interest for a topic.

Software developers are especially fond of Q&A websites, and a large number of these websites focus on different aspects of software development. The most prominent Q&A website aimed at developers is Stack Overflow, which hosts a large community of users and millions of questions about programming and technology. Stack Overflow currently receives over 120 million monthly visitors and ranks on the top 50 most popular websites in the world [1].

The large amounts of data generated through Q&A websites are an essential asset to understanding how the members of the communities behave, how they share knowledge among them, and how useful these websites are for users.

While there have been studies on other Q&A websites, software engineering researchers have invested enormous effort in studying Stack Overflow, and hundreds of papers have used its data since its initial release in 2008 [4]. Mamykina et al. [51] showed that Stack Overflow was more effective than other Q&A websites with high response rates and fast response times on the website’s

⁵ <https://answers.yahoo.com/>, accessed February 5, 2021.

⁶ <https://www.quora.com/>, accessed February 5, 2021.

first years online while. Asaduzzaman et al. [9] explored why questions go unanswered and Wang et al. [72] identified factors that lead to fast answers.

Several studies have examined the topics users ask about [5, 6, 12, 13, 65, 73], their opinions [47], pain-points [27], and their overall asking and answering behaviour [66, 70]. Others have analyzed the quality of the knowledge that is shared. Zhang et al. [81] analyzed obsolete answers, while Parnin et al. [57] have compared the website’s discussions to API documentation. Research has also shown the usefulness of the information shared on the website, exploring how developers use the code shared on the website [7, 77], and how to use it to augment other software development resources [31, 71].

Studies have also investigated methods of improving Stack Overflow, such as helping users find the information [56, 80], identifying expert users and using their knowledge to help the community [33], and increasing user contribution using symbolic rewards, such badges and bounties [19, 84].

Fewer studies explored the content of other Q&A communities. Hong et al. [41] have analyzed how users share knowledge on a health Q&A website, and Fu and Oh [35] analyzed the quality of answers on social Q&A websites. Other studies have provided ways of improving Q&A websites, such as increasing the overall politeness [74], increasing the popularity of academic answers [83], finding experts to answer questions [60, 68], predicting relevant discussions [61], and improving the way the community welcomes new users [67]. A significant portion of the research on other Q&A websites also focused on exploring what motivates users to participate in these communities [20, 28, 30, 36, 44, 82].

2.2 Game development Q&A websites

Among the many Q&A websites on the Internet, there are several that focus on discussing the development of games. Despite game development being composed of several skills ranging from programming to physics, these Q&A websites for game development usually revolve around a single game development engine and seek to help users find the solutions to the problems that surface when using that engine.

Game development engines are an integral part of the game development environment and shape the way games are made [34]. Unity and Unreal Engine 4 (UE4) are two of the most used game development engines, aggregating millions of registered users and being used to develop a large number of commercial games. These engines also ranked among the most popular technologies used by survey respondents on Stack Overflow’s 2020 developer survey [2].

Both Unity and UE4 provide specialized Q&A websites, namely Unity Answers and the UE4 AnswerHub, to help engine users find the answers to their game development questions. Unity Answers and the UE4 AnswerHub are the two largest Q&A websites specializing in game development, hosting hundreds of thousands of questions about this topic. In comparison, Stack Overflow has less than sixty-five thousand questions related to these game

engines, while the Game Development Stack Exchange hosts fifty thousand questions in total, as of February 2021.

Game engines have seen various applications in research, such as simulating rockfalls [39], running behavioural experiments [17], and exploring urban areas [26]. There have also been studies on their performance [22, 53] and their use in serious game development [22, 24]. Studies on the game development community itself tend to focus on social aspects, such as gender [15, 16, 32] and the relationship between developers and game engines [75], rather than user behaviour and how they share knowledge. As far as we know, our study is the first to explore game development Q&A websites and their communities.

2.3 Topic modeling of Q&A websites

The Latent Dirichlet Allocation (LDA) model is a statistical generative model that identifies latent topics present in the set of documents given as input using a Dirichlet distribution [14]. The model assigns each word a probability of belonging to a given topic, therefore allocating them into groups. We can then use these probabilities to classify documents into the topics represented by the words. The LDA model is unsupervised, taking only the corpus and the number of topics to be identified as input, along with other tuning parameters. The final topics are unlabelled, and we need to identify their meaning (if any) by manually analyzing the words that compose them.

Many studies have used LDA for modeling topics from corpora of documents extracted from different sources [43]. The LDA model has also seen ample use in software engineering and mining software repositories research, where researchers used the extracted topics to analyze discussions related to different aspects of software development [21]. For example, Ray et al. [63] used the topics extracted from feature descriptions on GitHub to identify project domains, while Lukins et al. [50] used an LDA-based approach for automatic bug localization in open-source software. Other have used LDA to categorize and detect duplicate bug reports [18, 69], and analyze reviews for mobile apps [42] and games [29].

Software engineering researchers have also amply used LDA to identify topics discussed on Stack Overflow. For example, Allamanis and Sutton [6] found that software developers discuss topics relating to how and why implementations and technologies do or do not work, while Barua et al. [12] analyzed how topics interact and change over time. Other studies analyzed the topics discussed on Stack Overflow about specific software development subjects such as machine learning [11, 37], mobile development [49, 65], and security [78]. Researchers have also analyzed the trends for topics of different software development disciplines finding that, for example, continuous engineering topics attracted fewer answers and are decreasing over time [79], while web development topics increased their share of questions [10]. We use similar techniques as those studies to analyze the topics of posts on Q&A websites

for game development. Our study is, to the best of our knowledge, the first to analyze the topics discussed by game developers in Q&A websites.

3 Methodology

We used five distinct datasets from different sources for this study. These datasets comprise data we collected from four Q&A websites and from the responses game developers gave to our survey about these communities. In this section, we describe the process of collecting and processing this data before using it for our analysis. Fig. 1 provides an overview of the steps taken in our methodology. The code and data used in this study can be found online in our replication package⁷.

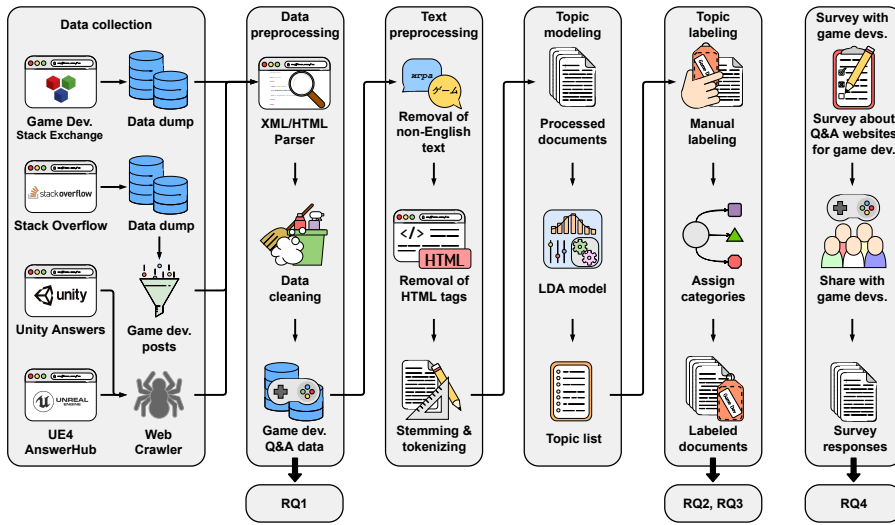


Fig. 1 Overview of our data collection and processing methodology.

3.1 Data collection

We used data from four Q&A websites in this study, namely Unity Answers, the UE4 AnswerHub, Stack Overflow, and the Game Development Stack Exchange. We chose to analyse these websites as they are the four largest Q&A websites discussing game development that we could find.

We acquired the data from Stack Overflow and the Game Development Stack Exchange from the December 2020 Stack Exchange data dump⁸. For

⁷ <http://doi.org/10.5281/zenodo.5047790>.

⁸ <https://archive.org/details/stackexchange>, accessed February 5, 2021.

Stack Overflow, we selected posts (i.e., questions, answers, and comments) that had at least one of the following tags to compose our game development dataset: *‘game-engine’*, *‘game-physics’*, *‘game-development’*, *‘gameobject’*, *‘2d-games’*, *‘unreal-engine4’*, *‘unreal-blueprint’*, *‘unreal-development-kit’*, *‘unrealscript’*, *‘unityscript’*, *‘unity-ui’*, *‘unity-webgl’*, *‘unity-networking’*, *‘unity-editor’*, *‘unity5’*, *‘unity5.3’*, *‘unity2d’*, *‘unity3d’*, *‘unity3d-5’*, *‘unity3d-mecanim’*, *‘unity3d-unet’*, *‘unity3d-2dtools’*, *‘unity3d-terrain’*, *‘unity3d-editor’*, *‘unity3d-gui’*. We manually selected these tags from the list returned by Stack Overflow’s tag search engine⁹ for the terms ‘game’, ‘unity’, and ‘unreal’. We only selected tags related to overall game development and game engines and excluded tags related to other frameworks and programming languages, such as Python and JavaScript.

As opposed to the other websites, Unity Answers and the UE4 AnswerHub do not share their data in a structured format. Therefore, we created a web crawler to harvest data directly from the Unity Answers and UE4 AnswerHub websites. We used the websites’ question catalogs, displayed on their homepage, as the starting point of crawling. These catalogs list the questions posted on the sites and contain IDs used to access them. We collected all of the question pages indexed by the catalogs, from oldest to newest. The pages also included the answers and comments posted as replies to the question. Likewise, we extracted the IDs of users that authored these posts and used them to access their profiles. We conducted the same procedure for Unity Answers and the UE4 AnswerHub. The crawler ran from August 3rd to August 11th, 2020, collecting 593,006 pages from Unity Answers and 279,902 pages from the UE4 AnswerHub.

3.2 Data preprocessing

The data we collected is in XML format for Stack Overflow and the Game Development Stack Exchange, and HTML format for Unity Answers and UE4 AnswerHub. We therefore parsed those files into tabular format and selected the information we use in this study. For all posts, we selected the post ID, post date, author ID, and their text contents. We also selected the fields indicating to which post an answer or comment belongs, and the fields indicating if an answer was accepted and a question was resolved. For Unity Answers and the UE4 AnswerHub, we collected information about the type of user (e.g., moderator or staff member).

We performed several cleaning steps on the data, such as type conversions, date formatting, and removal of posts with missing information. Table 1 shows a summary of our data at the end of the cleaning process.

⁹ <https://stackoverflow.com/tags>, accessed February 5, 2021.

Table 1 Summary of the data we collected from Q&A websites about game development.

Website	Questions	Answers	Comments	Posts'	Users
Unity Answers	381,055	347,264	652,091	1,380,410	196,954
UE4 AnswerHub	193,338	170,269	402,494	766,101	80,094
Stack Overflow	64,648	73,002	228,540	366,190	65,735
Game Dev. Stack Exchange	50,395	76,478	211,627	338,500	33,735
Total	689,436	667,013	1,494,752	2,851,201	376,481

'Posts = Questions + Comments + Answers

3.3 Text preprocessing

Prior to extracting topics from the posts for RQ2 and RQ3, we perform a series of text preprocessing steps. These steps aim to remove any of the text's unwanted features and provide better results in the next steps of the methodology.

We started by removing posts that belonged to any sections of the websites that were not primarily in English. We used a unique token to represent the presence of code in the texts, and replaced any code contained between the `<code>` HTML tag with it using regular expressions. Similarly, we replaced images using regular expressions that matched `` tags, and URLs using regular expressions that matched `<a>` tags or a contiguous string of characters preceded by `http://` or `https://`. We removed any of the other HTML tags or symbols using Python 3's BeautifulSoup 4 library¹⁰. Finally, we used Python 3's Gensim library¹¹'s `text_preprocess` function with default parameters to remove punctuations, multiple whitespaces, numeric characters, stopwords, and short words. The function also stems the texts using the Porter Stemmer [59], and tokenizes it by splitting words separated by spaces.

3.4 Topic modeling

We used two Latent Dirichlet Allocation (LDA) models to extract topics from the websites' posts. As Unity Answers and the UE4 AnswerHub are the official Q&A websites for their game engines, we believe the terminology used on them is different than that used on the other two websites. Therefore, we trained one of the models using data from Unity Answers and the UE4 AnswerHub, and the other one using data from Stack Overflow and the Game Development Stack Exchange.

We used Gensim's implementation of the algorithm provided by its `ldamodel` function. We set the `alpha` and `eta` parameters to "auto", the `passes` parameter to 1, and the `eval_every` parameter to 5. We also used a value of 42 for

¹⁰ <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>, accessed February 5, 2021.

¹¹ <https://radimrehurek.com/gensim/>, accessed February 5, 2021.

the random state to ensure reproducibility. We defined the number of topics as 30, as we believed that was a good number of topics for an initial analysis.

We used the preprocessed texts obtained in Section 3.3 as input for the model. As an output, the model produced 30 sets of words accompanied by their probabilities of belonging to each set for each post. These sets represent the topics identified by the model based on the initial parameters. We used the word probabilities to assign the posts to each topic.

3.5 Topic labeling

The topics provided by the LDA models were initially unlabeled, given that they are only collections of co-occurring words that do not necessarily carry any meaning with them. Therefore, we manually labeled each of the topics from both models by analyzing their 15 most important words. We also randomly sampled and read ten posts from each topic to corroborate our manual classification. We excluded two topics from Stack Overflow and the Game Development Stack Overflow, as we could not find a suitable label for them.

After labeling the topics, we found that some of them comprised a large portion of the posts while providing little information regarding their content. Six of the topics (three from each model) were related to words commonly used in Q&A discussions, such as “help,” “try,” and “issue,” and do not have a specific relation to game development. However, they are present in over 90% of posts and appear as the three most important topics in most of them. We have thus opted to ignore these topics for our analysis by unassigning them to their posts. Table 2 shows a list of all of the topics, including their most important tokens.

We note that eliminating the topics left some posts without a topic assigned to them. The absence of a topic should indicate that the post discusses no other topic of interest. We opted to remove those posts from our analysis, as they only accounted for 0.6% (16,979) of the total. Thus, after removing the topics mentioned above, we obtained 27 topics for Unity Answers and the UE4 AnswerHub, and 25 topics for Stack Overflow and the Game Development Stack Exchange.

We further classified the topics into three broad categories to analyze the higher-level concepts discussed in the posts. We thus assigned any topic related to a specific programming language, paradigm, or concept to the “General software development” category and topics discussing errors, crashes, and bugs related to any tool, framework, platform, or programming language to the “Bugs, crashes and errors” category. We assigned the rest of the game development topics to the “Game development” category.

3.6 Survey with game developers

We ran a survey with game developers from June 16th to August 14th, 2020. We shared the survey on major online game development communities on

Table 2 Summary of the labels and categories manually assigned to the topics obtained from running the LDA algorithm to game development Q&A data.

Websites	Category	Topic	Top 5 tokens
Unity Answers and the UE4 AnswerHub	Game development	Game objects	object, compon, spawn, attach, destroy
		Meshes	mesh, node, map, scale, size
		Lighting	light, normal, shadow, graph, dynam
		Rendering	ing, text, screenshot, imag, render
		Game mechanics	frame, block, speed, fp, damag
		Movement	rotat, move, direct, forward, movement
		Geometry	point, sphere, shot, cube, draw
		Game loop	time, run, second, continu, stop
		External tools	widget, plugin, thread, launcher, process
		Positioning	locat, pawn, cast, target, hit
		Collisions	collis, box, physic, overlap, wall
		3D modeling	import, model, templat, export, root
		Character animation	charact, plai, control, anim, state
		Materials	materi, textur, color, channel, bump
		Camera and display	camera, screen, world, main, view
Stack Overflow and the Game Development Stack Exchange	General software development	Game engines	project, engin, crash, build, file
		FX	sound, parent, child, item, hear
		Runtime	save, content, load, game, devic
		General programming	privat, float, public, int, playercontrol
		Networking	player, server, client, connect, send
		GUI	click, select, button, input, press
		Object Oriented Programming	class, type, instanc, static, custom
		Event handling	event, tick, enabl, disabl, true
		File management	editor, file, asset, addit, data
	Bugs, crashes and errors	Bug reports	issu, problem, fix, level, bug
		Programming errors	cpp, string, arm, format, lol
		General errors	error, compil, messag, fail, dll
	Game development	Game objects	object, script, compon, variabl, gameobject
		Rendering	render, textur, shader, mesh, map
		Game environments	scene, agre, disabl, particl, enabl
		Viewport	window, platform, devic, opengl, mode
		Game mechanics	player, tile, enemi, spawn, mechan
		Movement	right, direct, turn, left, move
		Geometry	point, object, angl, distanc, box
		Positioning	posit, rotat, vector, transform, calcul
		Collisions	collid, physic, collis, forc, hit
		3D modeling	model, import, format, blender, score
		Character animation	anim, charact, item, grid, hand
		Display	size, scale, draw, target, unit
		Camera	camera, space, screen, world, coordin
		Game engines	uniti, project, error, build, version
		Sound/audio	sound, plai, step, sourc, cube
	General software development	Runtime	data, save, load, text, store
		GUI	control, button, true, click, select
		General programming	valu, number, arrai, list, float
		User accounts	user, share, profil, pictur, cach
		Networking	server, client, send, connect, messag
		Publishing apps	support, app, googl, api, applic
		Java/Android development	main, thread, librari, coroutin, java
		Object Oriented Programming	type, class, instanc, entiti, static
		Event handling	input, state, event, action, inspector
		File management	imag, sprite, file, level, path

websites such as Facebook and Reddit, and received 347 responses. A list of the surveyed communities is available online in our replication package.

We asked the developers multiple-choice and essay questions about their use of Unity, UE4, and their Q&A communities. The survey comprised 48 questions in 20 sections¹². We directed respondents to different sections depending on their answers. For example, we asked questions about Unity only to those that used it for studying or working in the past.

All respondents also answered questions about their previous experience with game development and had the option to provide additional personal information such as their age and their country of residence. We used this information to characterize the population that originated our sample of re-

¹² Survey available at <https://figshare.com/s/a8fe3851e409967e2d39>

Table 3 Summary of the respondents of our survey.

	Category	Responses		
Age	≤17	20	(6%)	■
	18-29	223	(64%)	■
	30-39	80	(23%)	■
	≥40	23	(7%)	■
	Prefer not to say	1	(<1%)	■
Gender	Man	313	(90%)	■
	Woman	22	(6%)	■
	Other	5	(1%)	■
	Prefer not to say	7	(2%)	■
Game developer type (not mutually exclusive)	Hobbyist	141	(41%)	■
	Student	107	(31%)	■
	Professional	179	(52%)	■
Years of experience in game development	≤1	77	(22%)	■
	2-3	53	(15%)	■
	3-4	89	(26%)	■
	5-9	77	(22%)	■
	≥10	51	(15%)	■

spondents. Table 3 shows a summary of the responses we acquired for these questions. The majority of respondents are male game developers of less than 30 years of age. A significant portion of respondents (31%) are students, and many have only a few years of experience, which indicates that they are still early in their game development careers.

4 Results

In this section we discuss the motivation, approach and findings for each of our Research Questions.

4.1 RQ1. How did the studied game development Q&A communities evolve in terms of user participation?

Motivation: The goal of this research question is to uncover the trends experienced by the studied communities through a set of different measures. These trends provide insights on how successful the studied communities have been in nurturing and maintaining an active user base. We also identify a group of factors to investigate possible correlations to the changes we observed. Understanding what those factors are and how they relate to these trends allows us to derive actions for helping the communities prosper.

Approach: We used the creation dates of posts to analyze the evolution of the studied Q&A communities. We grouped the posts (questions, answers or comments) into months by removing the day from the post date. The monthly data provided less noise for temporal analysis. We removed the last incomplete month before the data collection, as it does not provide enough data for an accurate analysis. Overall, we grouped the posts into 131 months for Unity Answers (from October 2009 to July 2020), 78 months for the UE4 AnswerHub (from March 2014 to July 2020), 146 months for Stack Overflow (from August

2008 to December 2020), and 128 months for the Game Development Stack Exchange (from December 2009 to December 2020).

We used the number of total posts per month and the number of active users per month to measure community activity. We defined the number of active users as the number of unique users that created a post in a given month.

We analyzed the percentages of answered and resolved questions per month as a proxy for the communities' ability to help users who post questions. Answered questions are questions with one or more answers, while resolved questions are questions that have an accepted answer, indicating that the issue at hand was actually solved. We calculated these percentages by dividing the number of answered and resolved questions by the total number of posted questions in a month. Likewise, we measured the answer effectiveness per month, which we defined as the percentage of accepted answers from the total number of answers posted in a month.

We used the Cox-Stuart test [25] to obtain the statistical significance of the trends in these measures. With this test, we test the null hypothesis of randomness against the alternative hypothesis of a decreasing or an increasing trend in the data. The Cox-Stuart test divides the sequence of observations in half and compares them by performing a sign test. The test yields a p-value (p) that we used to discard the null hypothesis if less or equal to a threshold of 0.05.

We moved on to identify factors correlated to the trends observed using the steps above. We identified thirteen major releases related to the Unity and UE4 game engines that occurred during the studied period. We defined major releases as the release of new products or updates in which there was a change in the most significant digit of the version number of the game engines. For Unity, we also included the release of version 3.5, as it introduced several major new features and was described by their creators as "one of the biggest additions to Unity since its inception"¹³ at the time of its release. Major releases are associated with significant changes in the engines, which research has shown correlates to increased activity [48]. In addition, changes in the licensing models (such as the free and public releases for Unity and UE4) and the release of asset stores have made game development with those engines more accessible to the public, which may also lead to increased activity. Table 4 shows the list of releases we identified.

We compared the number of posts and active users in Unity Answers and the UE4 AnswerHub 30 days before and after each release to measure the correlation between these releases and the communities' trends. We only considered these two communities for the release-level analysis as they are focused on discussing a single engine each. All releases were more than 60 days apart, and the selected periods did not overlap. We left out three releases that occurred on the first or last 30 days of the observed period and did not provide a

¹³ <https://unity.com/our-company/newsroom/unity-technologies-releases-unity-3-5>, accessed June 28th, 2021.

large enough sample for this comparison. We measured the difference between the distributions of the two 30-day intervals using the median daily number of posts and active users. We ran the Wilcoxon-signed-rank test [76] to decide whether these differences were statistically different. We used the p-value (p) provided by the test to discard the null hypothesis of the two samples belonging to the same distribution if below a threshold of 0.05. This method is similar to the one used by Linares-Vásquez et al. [48] to identify the correlations between API changes and discussions on Stack Overflow.

We also calculated Cliff’s delta (d) to show the magnitude of this difference [23]. The delta describes the proportion of times that a median from the first period is higher than one from the second. We used the interpretation proposed by Romano et al. [64] to define the effect size as negligible ($|d| \leq 0.147$), small ($0.147 < |d| \leq 0.33$), medium ($0.33 < |d| \leq 0.474$), or large ($0.474 < |d| \leq 1$).

We calculated the Pearson correlation coefficient to analyze how user activity, as measured by the number of active users and posts per month, correlates to the changes in the other measures.

We used the total number of posts made by each user as a proxy for their experience in using the Q&A websites. Users start with zero experience and become more experienced as they interact with the community. Therefore, we calculated each user’s experience in each month after their first post to understand how user experience evolved.

Finally, we defined experienced users as the top 1% of users with the most overall experience in each community (1,875 users on Unity Answers and 802 users on the UE4 AnswerHub). Other studies also analyzed the contributions of the top 1% of the most experienced and active users on Stack Overflow [51, 55]. We measured the contributions of experienced users by the number of questions they answered and resolved and their answer effectiveness.

Findings: Three of the studied communities have become less active over the past few years. Fig. 2 shows the number of posts per month for the four studied websites. We observed that the number of posts per month decreases on the Game Development Stack Exchange after October 2012 ($p < .001$), on Unity Answers after March 2014 ($p < .001$), and on the UE4 AnswerHub after March 2015 ($p < .001$). These decreases occurred after periods of growth for Unity Answers ($p < .001$) and the Game Development Stack Exchange ($p = .03$), and stability for the UE4 AnswerHub ($p = .69$) after the communities’ initial release.

We found an 81% decrease when comparing the maximum number of posts in March 2014 (21,525) to the lowest one in November 2019 (4,093) on Unity Answers, and a 90% decrease on the UE4 AnswerHub, with the number of posts going from 21,736 to 2,273 between March 2015 and February 2020. In the Game Development Stack Exchange the number of posts dropped from 4,785 in October 2012 to 1,380 in September 2020, representing a 71% decrease.

We also observed the decline of those three communities (Unity Answers, the UE4 AnswerHub, and the Game Development Stack Exchange) in the number of active users per month (Fig. 3). We noticed that the decline only

Table 4 Major releases for Unity and the UE4 game engines. We only study releases related to Unity Answers and the UE4 AnswerHub, as these websites focus on discussing a single game engine each.

Engine	Label	Date	Description
Unity	a	28-10-2009	Free release
	b	15-10-2010	v3.0
	c	14-02-2012	v3.5
	d	13-11-2012	v4.0
	e	19-03-2014	Asset store release
	f	15-03-2015	v5.0
	g	10-07-2017	v2017.1
	h	02-05-2018	v2018.1
	i	15-04-2019	v2019.1
	j	22-07-2020	v2020.1
Unreal Engine	k	19-03-2014	v4.0/Public release
	l	03-09-2014	Marketplace release
	m	02-03-2015	Free release

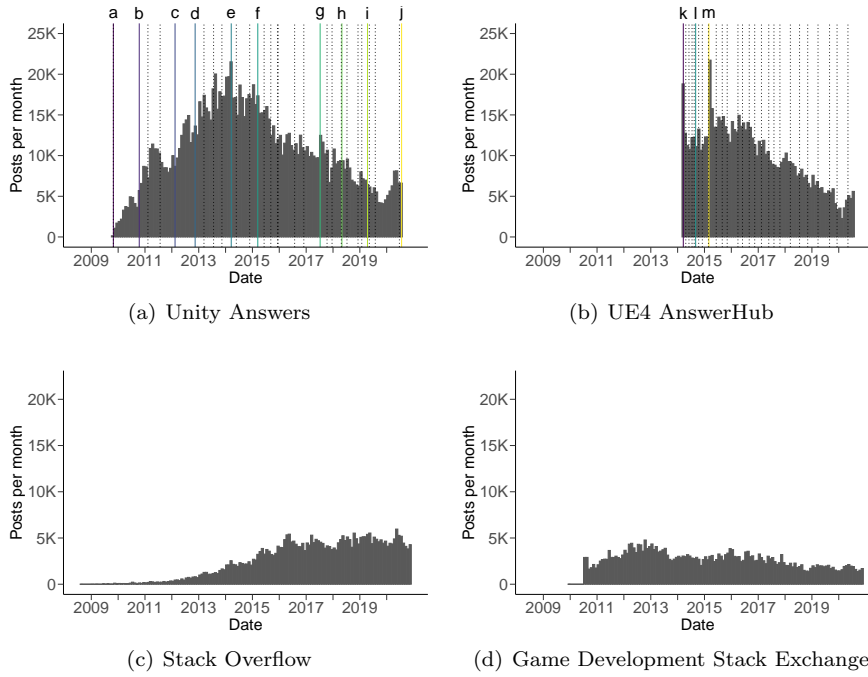


Fig. 2 Evolution of the studied Q&A communities given by the number of posts per month. The solid vertical lines represent events that occurred throughout the communities' lifetime. The line labels (*a-m*) refer to the ones shown in Table 4. Dashed vertical lines represent minor updates.

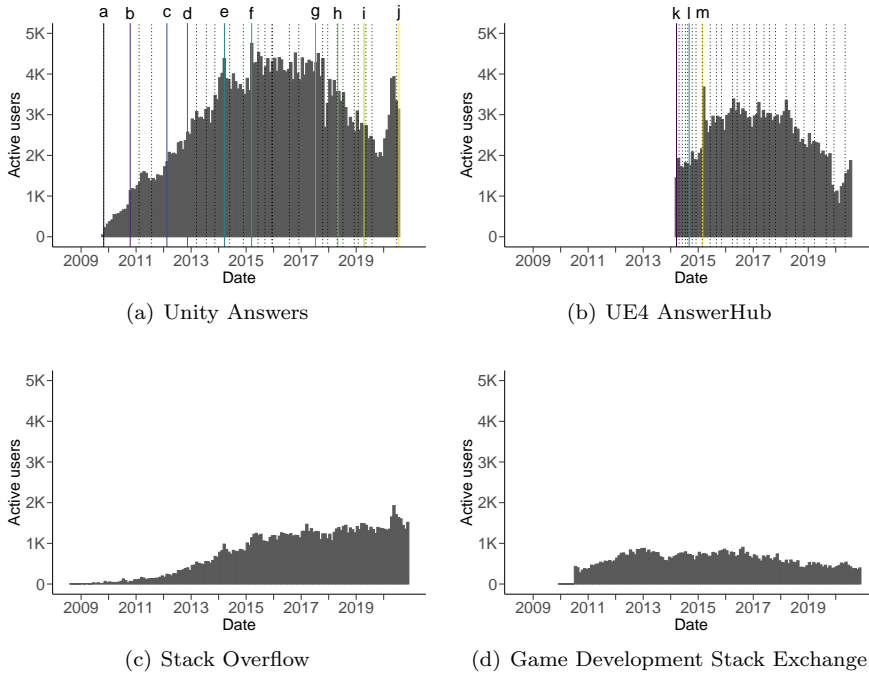


Fig. 3 Evolution of the studied Q&A communities given by the number of active users per month. The solid vertical lines represent events that occurred throughout the communities’ lifetime. The line labels (*a-m*) refer to the ones shown in Table 4. Dashed vertical lines represent minor updates.

started a few years after the decline in the number of posts per month in all of the communities. The three communities showed a similar pattern of early growth in number of active users followed by a period of stability before the decrease. On Unity Answers, the number of active users showed an initial growth until 2014 ($p < .001$) and a decreasing trend from late 2017 until early 2020 ($p < .001$). These trends are also present on the UE4 AnswerHub, where the number of active users grew until 2016 ($p < .001$) and started falling in 2018 ($p < .001$), and on the Game Development Stack Exchange, where that number grew until 2013 ($p < .001$) and decreased after 2016 ($p < .001$).

Between their highest and lowest points, the number of active users decreased 57% on Unity Answers, going from 4,481 to 1,953 between August 2017 and October 2019, 76% on the UE4 AnswerHub, going from 3,354 in March 2018 before to 816 in February 2020, and 61% on the Game Development Stack Exchange, going from 904 in August 2016 to 351 in October 2020.

We noticed that these measures increased on Unity Answers and the UE4 AnswerHub during the first months of 2020, especially after March. These trends are likely a symptom of the COVID-19 pandemic, which may have increased the time game developers dedicated for learning and improving their

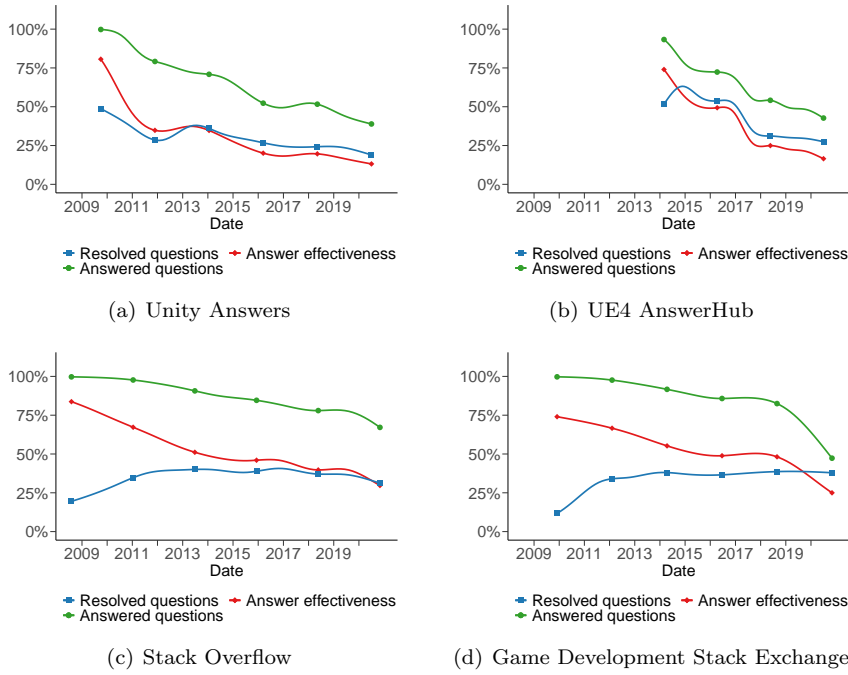


Fig. 4 Monthly percentages of answered and resolved questions, and answer effectiveness in the studied Q&A communities.

skills. The number of posts and active users reached levels similar to what they were in 2018. On Unity Answers, the growth seems to be short-lived, with considerable declines occurring in June 2020. However, we note that these increases are still in development, and more observations are needed to analyze their overall trends.

In contrast to those three communities, we did not observe any decreases in posts or active users per month on Stack Overflow (Fig. 2(c) and Fig. 3(c)). Instead, the number of game development questions and active users discussing them grew until 2016 ($p < .001$), and plateaued after that ($p = .09$). The fact that Stack Overflow did not suffer a decrease as the other communities may be a consequence of it not being dedicated to discussing game development topics, and being a popular place for asking general software development questions.

The percentages of answered and resolved questions decreased over time. Fig. 4 shows the percentages of answered and resolved questions for questions posted in each month in the studied communities. We found that in all of the communities, the percentages were at an all-time high during the first few months after the launch and decreased in most of the following months ($p < .001$ in all communities).

There were sharper declines on Unity Answers and the UE4 AnswerHub when compared to the other two websites. On Unity Answers, the percentage

of answered questions remained above 97% during the first twelve months, and reached its lowest point of 34% in June 2020. A similar decline occurred in the percentage of resolved questions, which peaked at 93% in October 2009 and reached a low of 11% in June 2020. On the UE4 AnswerHub, the percentage of answered questions fell from 97% in March 2014 to 37% in July 2020. Meanwhile, the percentage of resolved questions decreased from 81% in March 2014 to 14% in July 2020.

On the other hand, Stack Overflow and the Game Development Stack Exchange consistently kept the percentage of answered questions above 90% until 2013 and 2015, respectively. Furthermore, these percentages remained above 80% for most months until 2018, only showing steeper drops after that. Nevertheless, the percentage of answered questions reached 59% in November 2020 on Stack Overflow and 46% in April 2020 on the Game Development Stack Exchange. The percentages of resolved questions suffered similar decreases, going from 100% in October 2009 to 25% in November 2020 on Stack Overflow, and from 79% in September 2010 to 24% in September 2020 on the Game Development Stack Exchange. These percentages are higher than the ones found for all of the questions in Stack Overflow¹⁴, where the percentages of answered and resolved questions peaked at 88% and 65% in 2009, and had a low of 45% and 24% in October, 2020.

Answers have become less effective on Unity Answers and the UE4 AnswerHub throughout the years. Fig. 4(a) and Fig. 4(b) show the answer effectiveness on Unity Answers and the UE4 AnswerHub. We found that the answer effectiveness reached its highest value in the first months, peaking at 53% in February 2010 on Unity Answers and at 67% in December 2014 on the UE4 AnswerHub.

The answer effectiveness has fallen on Unity Answers when analyzing the whole studied period ($p < .001$), despite experiencing an increase between February 2012 and September 2013 ($p < .001$). The lowest effectiveness we found on Unity Answers was 18% in June 2020. On the UE4 AnswerHub, the answer effectiveness increased from March 2014 to December 2014 ($p = .03$), but decreased over the following years ($p < .001$), reaching a low of 23% in June 2019.

We observed different patterns on Stack Overflow and the Game Development Stack Exchange. On those communities, the answer effectiveness grew until 2013 ($p = .005$ for Stack Overflow and $p < .001$ for the Game Development Stack Exchange) before stabilizing in the following years. Furthermore, on Stack Overflow the answer effectiveness started decreasing in 2017 ($p = .009$).

The communities grew as major releases occurred. We found that the release of Unreal Engine 4 for free in March 2015 correlates to an increase in the number of posts per day ($p < .001$, large effect size) and of active users per day ($p < .001$, large effect size) on the UE4 AnswerHub. When comparing the

¹⁴ <https://data.stackexchange.com/stackoverflow/query/1284342/answered-and-resolved-questions#graph>, accessed February 5, 2021.

intervals prior and after the release, the median number of posts (456.5) and active users (212) per day increased over 50% (to 709 and 318.5, respectively).

On Unity Answers, a 35% increase in the median number of posts per day (from 146 to 197.5, $p = .003$, large effect size) and a 29% increase in the median number of active users per day (from 76.5 to 98.5, $p = .004$, large effect size) followed the release of Unity v3.0. The release of Unity v4.0 correlates to a smaller increase of 17% in the median number of active users per day (from 175.5 to 206, $p = .008$, large effect size).

These are only three of the ten major releases we analyzed. On Unity Answers, the release of the Unity Asset Store correlates to decreases of 19% (from 306.5 to 279.5, $p < .001$, large effect size) and 7% (from 728 to 592.5, $p = .02$, medium effect size) in the median number of posts per day and active users per day, respectively. Other major releases showed no statistically significant difference based on the test results. Nevertheless, the statistically significant releases correspond to the net growth in the communities. These results are consistent with the ones found for Stack Overflow, where changes in Android APIs correlate to increases in discussions [48].

The percentages of answered and resolved questions went down as the communities grew. The increase in the number of posts and active users in all of the communities correlates to a decrease in the percentages of answered and resolved questions. We found correlation coefficients below -0.5 between these measures during the period of the communities' growth.

After the growth, we found correlations between 0.4 and 0.9 for all of the communities but Stack Overflow. For Stack Overflow, the correlations between the percentages and the number of posts remained at -0.5 during the period of stability after 2016. We found no correlation between the number of posts and the percentages of answered and resolved questions for Stack Overflow after 2016.

The decrease of the percentages while the community grew may be an indication of an overload caused by the large number of posts. On the other hand, the effects of a smaller and less active community may be playing a part in decreasing these percentages during the communities' decline.

Experienced users stopped contributing to the communities. Fig. 5 shows the experience of active users over time on the studied communities. The overall experience of users started decreasing after 2013 on the Game Development Stack Exchange, after 2015 on Unity Answers and after 2017 on Stack Overflow and the UE4 AnswerHub. On the UE4 AnswerHub, 44% (34) of experienced moderators also became inactive in 2017.

This decrease occurred as experienced users become inactive, and new inexperienced users joined the community. While experienced users stopped using the communities, new users did not acquire enough experience to replace them.

Experienced users increased the percentage of answered and resolved questions. Experienced users significantly contributed to the communities by reducing the number of unanswered questions. We found that the top 1% most experienced users posted answers on 63% (149,794) of the 235,900 answered questions on Unity Answers, 67% (85,667) of the 128,100 answered

questions on the UE4 AnswerHub, and 60% (26,157) of the 43,799 questions on the Game Development Stack Exchange. Experienced users contributed slightly less on Stack Overflow, posting answers on only 43% (22,142) of the (51,971) answered questions.

Experienced users also provided more effective answers. On Unity Answers, these users had an answer effectiveness of 39%, which is 18 percentage points higher than the effectiveness of other users (21%). On the UE4 AnswerHub experienced users had an answer effectiveness of 57%, compared to the effectiveness of 35% of others. This difference was smaller on the Game Development Stack Exchange, where experienced users had an effectiveness of 41% (against 30% of other users), and on Stack Overflow, where experienced users had an effectiveness of 48% (against 32% of other users).

Overall, experienced users resolved 66% (68,588) of the 103,723 resolved questions on Unity Answers, 68% (54,833) of the 80,749 resolved questions on the UE4 AnswerHub, 54% (14,484) of the 26,806 resolved questions on the Game Development Stack Exchange, and 43% (11,812) of the 27,460 on Stack Overflow.

We found that 10% (77) of the most experienced users we selected on the UE4 AnswerHub were moderators. These moderators had an even higher answer effectiveness (80%), answered 38,937 questions (30% of the total number of answered questions), and resolved 32,540 questions (40% of the total number of resolved questions). We also identified moderators among the most experienced users on Unity Answers, but their contribution to the community was far smaller than that of the UE4 AnswerHub’s moderators.

Other studies have also found that a small percentage of experienced users contribute to Q&A communities by providing a large number of higher-quality answers [51, 55].

Summary: All of the studied communities have stopped growing during the last few years. The communities have become less active, and the percentage of unanswered questions has increased. Additionally, answers have become less effective, and most questions go unresolved. These changes correlate to the release of new products and engine versions, and to the decrease of the overall experience of the communities.

4.2 RQ2. What topics are most frequently discussed by game developers on the studied Q&A websites?

Motivation: Developing games requires a set of skills that are unique to the game development context [45, 58]. For example, other than the programming concepts needed to create and manipulate game behaviour, game developers also need a basic understanding of physics, graphic design, and sound editing. In this research question, we explore the topics of posts collected from the studied game development Q&A websites to identify how prominent they are in the community and how they have changed over time. This analysis reveals

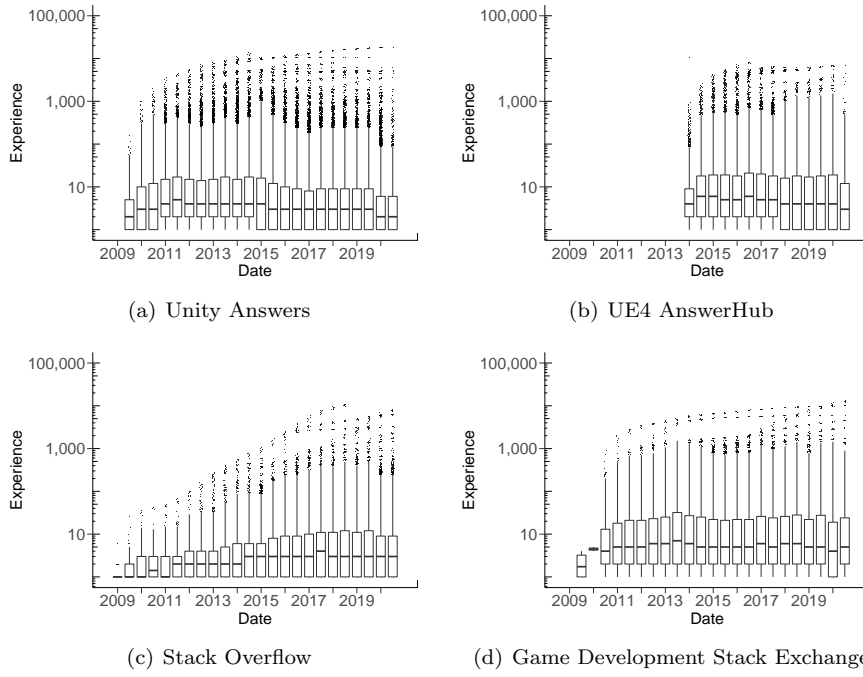


Fig. 5 Distribution of active user experience per semester (measured in the cumulative number of posts made by each user) for the studied Q&A communities.

the topics in which game developers are most or least interested or those with which they require more or less help in understanding. Game development websites and their communities can use our findings to identify which topics they must focus on to help game developers in their work and improve their experience.

Approach: We analyzed the distribution of posts per topic and category in each of the studied communities. We identified the all-time most discussed topics and categories according to the proportion of posts assigned to them. We chose to analyze the proportion instead of the absolute number of posts as we are interested in exploring how these topics relate to each other. Furthermore, by using the proportion of posts, we avoid the effect of the growth or decline of the game development community.

We compared the distributions of posts belonging to each topic in the studied websites. As we obtained the topics from two different LDA models (see Section 3.4), we compared Unity Answers and the UE4 AnswerHub separately from Stack Overflow and the Game Development Stack Exchange. We used the Chi-square test to determine if there is a statistical significant difference between the number of posts per topic for each pair of websites. We rejected the null hypothesis of the distributions being the same if the resulting p -value is below the threshold of 0.05.

While analyzing the topics in an all-time manner is useful for understanding historical and consolidated patterns, this approach hides possible trends that might have occurred over the years. The proportion of topics might change as time passes, and a long history of similar patterns might overshadow those that occurred in brief intervals. Thus, we use the time of posting collected with each post to calculate the distribution of topics per month.

We use the Cox-Stuart test [25] to identify any trend in the percentage of posts per month for each topic. We test the null hypothesis of randomness in the data against the hypothesis of non-randomness in the form of a decreasing or increasing trend. We then reject the null hypothesis if the resulting p -value is below a threshold of 0.00096 (0.05/52) that we obtained using the Bonferroni correction for 52 comparisons (one for each topic in Table 2).

Findings: Most of the topics discussed on the studied Q&A websites are specific to game development. Table 5 shows the number of topics belonging to the categories we defined. We found that 17 topics (63%) on Unity Answers and the UE4 AnswerHub are specific to game development, while 7 (26%) relate to general software development, and 3 (26%) relate to bugs, crashes, and errors. We found similar results on Stack Overflow and the Game Development Stack Exchange, with 16 topics (64%) being specific to game development, and 9 (36%) relating to general software development.

Table 5 Distribution of topics and posts per category.

Website	Category	Topics	% of topics	Posts	% of posts
Unity Answers	Game development	17	63%	795,506	58%
	General software dev.	7	26%	207,053	15%
	Bugs, crashes and errors	3	11%	375,248	27%
UE4 AnswerHub	Game development	17	63%	423,439	56%
	General software dev.	7	26%	96,116	13%
	Bugs, crashes and errors	3	11%	233,383	31%
Stack Overflow	Game development	16	64%	232,702	64%
	General software dev.	9	36%	133,048	36%
Game Development Stack Exchange	Game development	16	64%	215,635	64%
	General software dev.	9	36%	122,092	36%

Table 5 also shows the number of posts assigned to each of the categories. We observed that 27% of posts on Unity Answers and 31% of posts on the UE4 AnswerHub belong to *Bugs, crashes and errors*, despite that category having only three topics assigned to it. On the other hand, only 15% of posts on Unity Answers and 13% on the UE4 AnswerHub belong to the *General software development* category. We did not identify any topics and posts relating to bugs, crashes and errors on Stack Overflow and the Game Development Stack Exchange, probably because these topics are specific to a game engine.

The distributions of topics show no statistically significant difference between the websites. Fig. 6 shows the distribution of topics on the studied communities. We found a p -value of 0.24 after comparing both pairs of websites using the Chi-square test, and thus could not reject the null hypothesis of those samples coming from different distributions.

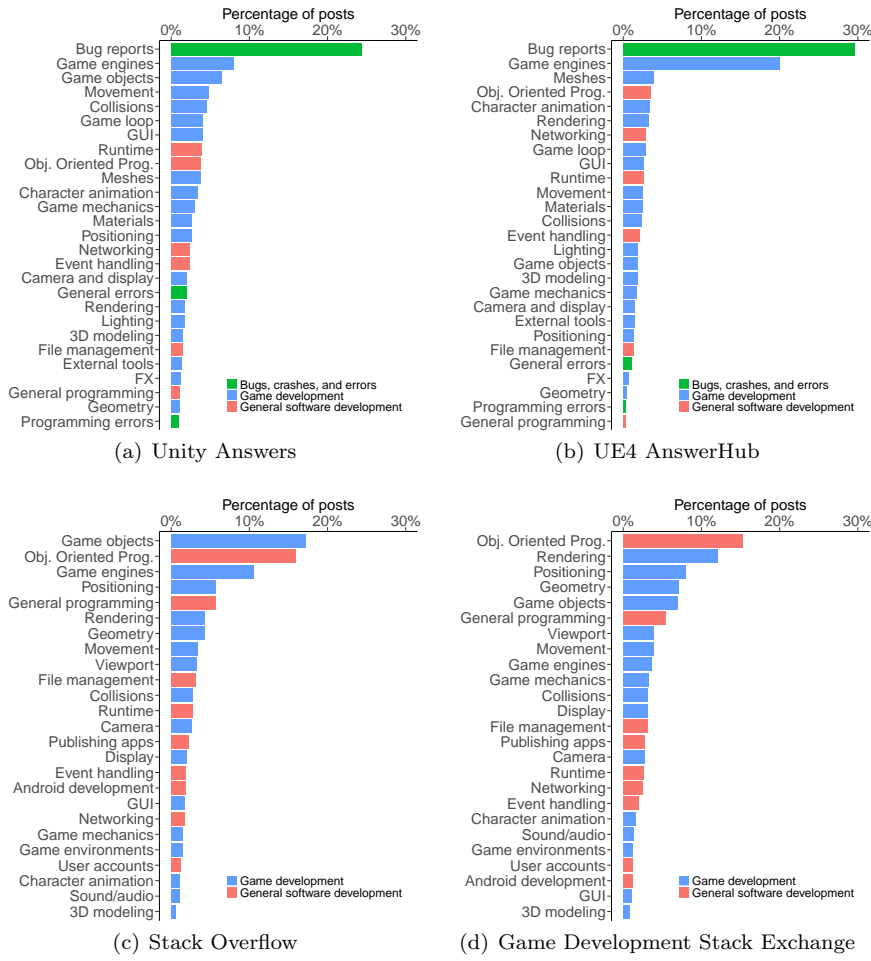


Fig. 6 Percentage of posts assigned to each topic in the studied Q&A communities.

However, we still observed slight differences in the ranking of topics between the websites. For example, we found that *Game objects* was among the least discussed topics on the UE4 AnswerHub, while Unity Answers frequently discussed it, given that Game Objects are a specific type of object used as a basis for building characters, props and scenery in the Unity game engine. Similarly, Stack Overflow had a larger focus on discussing *Game objects* than the Game Development Stack Exchange. Those differences indicate that the websites have slightly different focuses, which may be a consequence of the different engines discussed on them and overall community preferences.

Bug reports was the most discussed topic on Unity Answers and the UE4 AnswerHub. Fig. 6(a) and Fig. 6(b) show the percentage of posts assigned to each topic on Unity Answers and the UE4 AnswerHub. We found

that over 24% of posts on Unity Answers (334,839) and over 29% of posts on the UE4 AnswerHub (222,579) discuss *Bug reports*. Most of the other topics in those websites had fewer than 10% of posts assigned to them, with the only exception being *Game engines* on the UE4 AnswerHub. This finding indicates that game developers used Unity Answers and the UE4 AnswerHub as bug reporting platforms, as they are maintained by the companies that develop the game engines they discuss.

Game developers discussed different aspects of game development. Fig. 6 shows the distribution of topics on the studied communities. We observed that topics related to game development are among the most discussed in all of the communities and describe many different aspects unique to game development. For example, topics such as *Meshes*, *Rendering*, and *Positioning* relate to game world-building, while *Movement*, *Collisions*, and *Character animation* discuss game behaviour. Other topics discuss the tools used for game development (e.g., *Game engines*), and the implementation of various game mechanics (e.g., *Game loop*).

Other game development topics have accrued a smaller number of posts and do not appear among the top-ranked ones. The lower-ranked topics may represent the ones in which game developers are less interested or have fewer issues. We also observed that some of these topics relate to specific niches inside the game development community, such as *3D modeling*, *Sound/audio*, and *FX*, and may therefore have a smaller group of developers discussing them.

Object-oriented programming was one of the most popular topics on Stack Overflow and the Game Development Stack Exchange. Fig. 6(c) and Fig. 6(d) show the distribution of posts per topic on those two websites. We found that *Object-oriented programming* was the most discussed topic on the Game Development Stack Exchange with 15% of posts (51,671), and the second most discussed on Stack Overflow with 16% of posts (58,430). The *Game objects* topic was also popular on those websites, being the most frequently discussed on Stack Overflow with 17% of posts (62,934), and the fifth most discussed on the Game Development Stack Exchange with 7% of posts (23,586). Game objects are the base class for entities in the Unity game engine, and game developers need to understand basic object oriented programming concepts to use them effectively.

Game developers also discussed *Object-oriented Programming* and *Game Objects* in the other websites, although in smaller proportions. Together, the two topics accounted for only 10% of posts on Unity Answers, and 5% on the UE4 AnswerHub.

The percentage of posts changed over time for several topics. Table 6 shows the number of topics with increasing and decreasing trends ($p < 0.00096$) for each of the studied communities. Over 30% of the topics showed changes in all of the communities, and over half on Unity Answers and the UE4 AnswerHub. We noticed that abrupt changes occurred at the start of the studied period, which can be explained by the low number of questions asked in the communities at that time. However, most of the topics had subtle changes of only a couple percentage points over the years.

Table 6 Distribution of topics and posts per category.

Website	Increasing topics	Decreasing topics	Topics changed	% of topics
Unity Answers	6	9	15	56%
UE4 AnswerHub	13	2	15	56%
Stack Overflow	8	2	10	40%
Game Development Stack Exchange	4	4	8	32%

Summary: Game developers discussed topics specific to game development more frequently than others. These topics covered several aspects unique to game development. The most discussed topic on Unity Answers and the UE4 AnswerHub was *Bug reports*, indicating that game developers used those websites as official bug reporting systems. Game developers also frequently discussed object-oriented programming, especially on Stack Overflow and the Game Development Stack Exchange.

4.3 RQ3. What are the characteristics of posts from each topic?

Motivation: In RQ2 (Section 4.2) we explored and compared the topics discussed in the studied game development Q&A communities using the distribution of posts per topic and category. While that analysis is important for identifying relevant topics and trends in the game development community, we still lack understanding of how these topics differ in practice. For example, game developers might discuss some topics frequently but in little detail, which may be of little use for those seeking to learn more about the subject. On the other hand, unpopular topics might offer more in-depth discussions, which provide a more reliable source of information. Therefore, analyzing the characteristics of the posts from each topic and how they differ might uncover the topics at which the game development community is proficient and those it should focus on improving, while also setting expectations for newcomers.

Exploring how topics differ among themselves is useful for understanding the game development community as a whole. However, we have collected texts from posts of four distinct Q&A websites which have different features, communities, and focus on specific aspects of game development. Therefore, we also explore how those topics differ in terms of the websites where they are discussed. A better understanding of the websites' specific characteristics can aid game developers in choosing with which community they will interact and be a part of based on their preferences.

Approach: We compared posts from each topic in each community based on six different aspects. We compared questions, answers and comments separately, as each has unique characteristics ensuing from their distinct uses in the websites. Table 7 describes the 10 comparisons we made (5 for questions, 3 for answers, and 2 for comments) based on the type of the posts, along with brief descriptions for each of them. The aspects we chose for these comparisons serve as proxies for measuring the quality of discussions for each topic. For example, the number of responses and answered questions point to how much

help a user gets when asking a question. Meanwhile, the number of resolved questions and accepted answers and the length of post texts gauge the quality of the help provided by other users by measuring if correct solutions are provided and how thoroughly they are discussed. Finally, the presence of code in posts indicates if the topics are discussed at a more technical or conceptual level.

Table 7 Description of the comparisons made for each topic based on the type of post being compared.

Post aspect	Post types	Description
Number of responses	Questions	The number of responses (comments or answers) added to a question
Resolved questions	Questions	Whether the questions received an answer marked as accepted
Answered questions	Questions	Whether the question received at least one answer
Accepted answers	Answers	Whether the answer is the correct solution to a question
Length of text	Questions, Answers, Comments	The length of the text contained in the post in characters (excluding code)
Presence of code	Questions, Answers, Comments	Whether the post contains a code snippet

We chose to compare one topic against the rest as opposed to all-vs-all, as we sought to identify the ones that stand out from the norm. The comparison between two topics may be relevant when validating a specific hypothesis about them, but its meaning becomes harder to interpret when analyzed as one of many other comparisons. We compare topics separately for each of the four studied websites, to avoid any bias caused by the differences between the communities. We also compared the high-level categories we identified in Section 3.5, but decided to leave them out of our analysis as they mirrored the results we found for individual topics. Therefore, we perform 1040 comparisons ($27 \text{ topics} \times 10 \text{ comparisons}$, for Unity Answers and the UE4 AnswerHub, and $25 \text{ topics} \times 10 \text{ comparisons}$ for Stack Overflow and the Game Development Stack Exchange).

We used the Mann-Whitney U-test [52] to identify statistical significant differences between the distributions used in the comparisons. In this test, we reject the null hypothesis of two independent samples belonging to the same population if the resulting p -value is below a predefined threshold. To counteract the effect of multiple comparisons, we used the Bonferroni correction for 1040 comparisons, obtaining a threshold for the p -value of 4.8×10^{-5} .

Furthermore, we calculated the effect size of the statistical significant differences using Cliff’s delta (d). We used the same thresholds as in RQ1 (Section 4.1) to determine if the difference was negligible ($|d| \leq 0.147$), small ($0.147 < |d| \leq 0.33$), medium ($0.33 < |d| \leq 0.474$), or large ($0.474 < |d| \leq 1$), following the interpretation proposed by Romano et al. [64].

We also compared the values of each aspect to provide a more palpable comparison. For aspects that represent binary values, such as resolved questions and accepted answers, we calculated the percentage of posts showing a positive value for that aspect. We compared other aspects using their median values.

We also used the aspects described in Table 7 to compare each topic across different websites. However, as opposed to the one-vs-rest comparison used previously, we compared the distribution of a topic in one of the websites with the distribution of the same topic on the other. We still performed 520 comparisons, resulting from the comparison of the distribution of 10 post aspects for each of the 27 topics in Unity Answers and the UE4 AnswerHub, and 25 topics in Stack Overflow and the Game Development Stack Exchange.

Once again, we used the Mann-Whitney U-test [52] to determine the statistical significance of the differences, using the same threshold of 9.6×10^{-5} for its p -value obtained from the Bonferroni correction for 520 comparisons. We also calculated the effect size of the differences using Cliff's delta (d), and classified the results using the aforementioned criteria. Lastly, we compared the values of the aspects using the same percentages and medians described before.

Results: Most topics are significantly different. However, most differences are negligible. Table 8 shows a summary of the comparisons we performed for each aspect and each topic in the studied websites. Of the 1040 comparisons we performed, 676 (65%) showed statistical significant differences, meaning that the topics are different in most of the aspects we elected for comparison. Yet, we found that only 90 (13%) of these differences were non-negligible, and only four showed effect sizes above small.

Similarly, we found that topics differed in many aspects when comparing them across websites, but most of the differences were negligible. Out of the 520 comparisons we made, 383 (74%) were statistically significant. However, only 76 (20%) of these showed non-negligible effect sizes, 24 of them showing effect sizes above small. Our following analysis focuses on the non-negligible differences between topics.

The frequency in which game developers discussed code varied for some topics. Table 9 shows the topics with statistically significant and non-negligible differences in terms of presence of code in their posts studied communities. Nine of the topics we analyzed had more code in their posts than others. Notably, game developers discussed code more often in topics such as *Game Objects*, *Object-oriented programming*, and *General programming*. We note that these three topics relate to software development and programming concepts, which explains their higher percentage of code.

Positioning, *Movement*, and *Game loop* were the only other topics specific to game development to show a difference in terms of the percentage of posts discussing code in them. The presence of code indicates that those topics are focused on discussing implementation issues, while the rest of the game development topics discuss those subjects more conceptually.

Table 8 Summary of the comparisons we performed for each of the studied websites.

Website	Comparisons	Stat. significant	%	Non-negligible	%
Unity Answers	270	234	87%	35	15%
UE4 AnswerHub	270	189	70%	27	14%
Stack Overflow	250	131	52%	14	11%
Game Development Stack Exchange	250	122	49%	14	11%
Total	1040	676	65%	90	13%

Table 9 Topics with statistically significant and non-negligible differences with others in terms of presence of code in posts in the studied communities.

Topic	Post type	Website	Code (%)	
			In topic	In others
Game objects	Questions	Unity Answers	56	38
		Stack Overflow	91	61
		Game Dev. Stack Exchange	86	41
	Answers	Unity Answers	49	34
		Stack Overflow	92	44
		Game Dev. Stack Exchange	89	41
General programming	Questions	Stack Overflow	83	66
		Unity Answers	84	39
	Answers	Unity Answers	62	35
	Comments	Unity Answers	33	10
Object Oriented Programming	Questions	Unity Answers	63	39
		UE4 AnswerHub	33	10
	Answers	Unity Answers	56	34
		UE4 AnswerHub	27	7
General errors	Questions	Unity Answers	63	39
		UE4 AnswerHub	39	10
	Answers	UE4 AnswerHub	23	7
Positioning	Questions	Unity Answers	59	39
		Game Dev. Stack Exchange	63	45
	Answers	Game Dev. Stack Exchange	56	38
Movement	Questions	Unity Answers	56	39
	Answers	Unity Answers	49	34
Game loop	Questions	Unity Answers	56	39
File management	Answers	UE4 AnswerHub	23	8
Programming errors	Answers	UE4 AnswerHub	30	8

Many topics differ in the length of the text in all post types.

Table 10 shows the topics with statistically significant and non-negligible differences in terms of the median number of characters in questions and answers in the studied communities. 63 (70%) of the 90 non-negligible differences we found among topics were in terms of the length of the text contained in their posts. 37 of the differences occurred in terms of comments, 23 in terms of answers, and 3 in terms of questions. Those differences occurred in 21 different topics, with 9 of the topics only showing differences for comments. All of the topics had longer post contents than the others.

Some topics were longer in multiple websites and post types, such as *Networking*, *Collisions*, *Object Oriented Programming*, and *Positioning*. We found the largest differences between topics on the Game Development Stack Ex-

Table 10 Topics with statistically significant and non-negligible differences with others in terms of the number of characters in their questions and answers in the studied communities.

Topic	Post type	Website	Median length Topic	Others
Networking	Questions	Game Dev. Stack Exchange	828	636
		Stack Overflow	654	545
		Game Dev. Stack Exchange	869	558
	Answers	Stack Overflow	430	320
		UE4 AnswerHub	321	253
		Unity Answers	288	226
Object Oriented Programming	Questions	Game Dev. Stack Exchange	761	636
	Answers	UE4 AnswerHub	308	252
		Unity Answers	291	224
Collisions	Answers	Stack Overflow	397	319
		UE4 AnswerHub	330	253
		Unity Answers	310	222
Positioning	Answers	UE4 AnswerHub	315	254
		Unity Answers	282	225
Materials	Answers	UE4 AnswerHub	314	253
		Unity Answers	286	225
Game mechanics	Answers	Game Dev. Stack Exchange	908.5	556
		Unity Answers	291	225
Game objects	Answers	UE4 AnswerHub	305	253
		Unity Answers	290	221
Movement	Answers	UE4 AnswerHub	319	253
		Unity Answers	317	223
Geometry	Answers	Unity Answers	292	226
Lighting	Answers	UE4 AnswerHub	353	253
FX	Answers	UE4 AnswerHub	335	254
Character animation	Answers	UE4 AnswerHub	316	253

change, where the median character length of answers about game mechanics and networking was 908.5 and 869, respectively, compared to the median of 556 of others. The length of the text of a post may reflect the complexity and richness of the description used in the discussion. The fact that game developers provide more detail when discussing *Networking* and *Collision* may increase the effectiveness of the discussion by providing better answers for questions [38]. On the other hand, shorter questions may have superior quality than others [62] and may receive more useful answers [54].

Posts on Unity Answers and Stack Overflow discussed code more frequently than on the UE4 AnswerHub and the Game Development Stack Exchange. We observed that posts in Unity Answers contained code more frequently than in the UE4 AnswerHub for 23 topics in 46 comparisons. The differences were small for 27 comparisons, medium for 15 comparisons, and large for 4 comparisons. The largest differences between the websites occurred in the *General programming* topic, with 84% of questions and 62% of answers on Unity Answers containing code, against 61% and 43% on the UE4

AnswerHub. The fact that the UE4 AnswerHub discussed less code may be an effect of its use of the Blueprints Visual Scripting system, which replaces the need of writing raw code in many tasks. The Unity engine did not have a visual scripting tool until August 2020.

We observed similar results when comparing Stack Overflow with the Game Development Stack Exchange. We found that 20 topics had more code on Stack Overflow, with small differences for 25 comparisons and medium differences for 5 comparisons. This time, the largest differences occurred in the *Game Mechanics* topic, with 76% of questions and 50% of answers containing code on Stack Overflow, against 37% and 17% on the Game Development Stack Exchange.

Summary: The topics showed significant differences in many aspects across the studied communities. Despite most of the differences being negligible, we could still identify topics that differed with at least small effect sizes in terms of the percentage code snippets in their posts and the length of their text. We also found significant non-negligible differences between posts belonging to the four websites, mainly in terms of the percentage of posts containing code.

4.4 RQ4. How do game developers perceive the studied communities?

Motivation: The data we collected from the studied Q&A communities allowed us to quantitatively study them through the interactions of users, and the tools provided by their websites. Our previous analyses are limited by the information we can infer from these interactions, and our previous findings build an incomplete image that may not reflect what developers undergo when using these communities. In this research question, we bridge the gap between our previous analysis and the actual way users experience the communities.

Approach: We used the 347 responses we collected from our survey with game developers (Section 3.6) to study their perception of Unity Answers and the UE4 AnswerHub. We chose to ask questions only about these two websites, as they each have a unique community associated with their game engine. Only survey respondents who reported having used Unity or UE4 for studying or working answered questions about that engine's community. We further divided that group of respondents to isolate the ones that had actually used the studied Q&A websites.

We analyzed the frequency with which engine users accessed the Q&A websites. We split respondents into three mutually exclusive categories according to their access frequencies: those that had never accessed the site, those that had at least once, and those that did it regularly. We defined regularly as one or more accesses per month. Fig. 7 represents the funnel that respondents went through when answering the survey, which allowed us to separate them into these groups.

Respondents who had accessed at least one of the Q&A websites indicated if they used the communities to ask questions, answer questions, report bugs, or search for a question related to theirs. We gave these respondents the option to write what they liked and disliked about the communities.

We asked respondents who regularly accessed the Q&A websites what the highest frequency with which they accessed the websites was. We also asked these respondents to classify their access as daily, weekly, or monthly. If their access frequencies changed, respondents chose their current frequency among those same options, with the addition of “less than monthly” and “stopped accessing.” Respondents who changed their access frequencies or did not regularly access the websites could provide a reason for doing so.

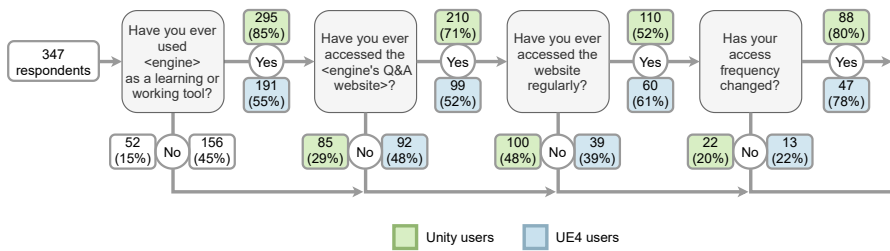


Fig. 7 Representation of the selection process respondents went through when answering the questions of our survey. The boxes show the number of respondents which answered each of the questions with “Yes” or “No”.

We compared the respondents’ usage of the Q&A communities to their usage of a set of other learning resources. We identified twenty resources available for Unity and nineteen for UE4. We asked engine users to indicate the frequency with which they accessed each resource for finding solutions to their game development questions. We split respondents according to their access frequency of these resources using the same categories we did for the Q&A websites.

Findings: Respondents prefer other resources for finding solutions to their game development problems. Fig. 8(a) and Fig. 8(b) show the access categories of respondents who used Unity and UE4 for the most frequently accessed learning resources, along with Unity Answers and the UE4 AnswerHub. We found that most respondents are aware of the existence of the studied Q&A communities. 71% (210) of 295 Unity users accessed Unity Answers, and 52% (99) of 191 UE4 users accessed the UE4 AnswerHub at least once in the past. These percentages are low when compared to the ones we found for other resources. A higher percentage of respondents reported having used the engines’ documentation (93% of Unity users, 87% of UE4 users), video tutorials (93% and 89%), online guides (85% and 82%) and the engines’ official forums (86% and 79%). A high percentage of Unity users (84%) also report having used Stack Overflow in general, likely looking for solutions related to the C# programming language used in the Unity engine. Considering the

percentage of respondents that used each resource, Unity Answers ranked seventh out of the twenty resources we identified for Unity. The UE4 AnswerHub ranked eleventh out of the nineteen resources we identified for UE4.

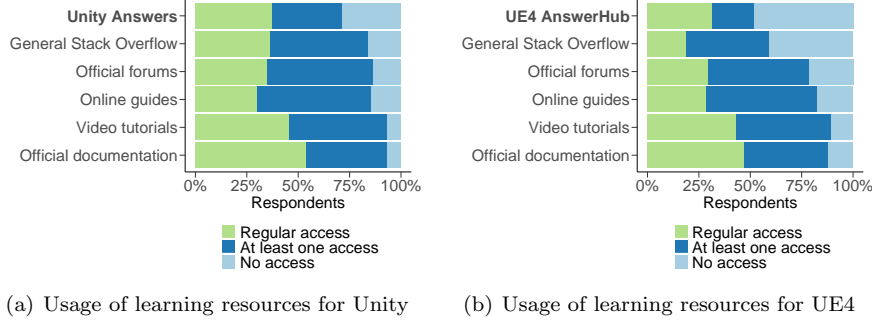


Fig. 8 Access categories of Unity and UE4 users for the most frequently accessed learning resources.

In contrast, respondents accessed Unity Answers and the UE4 AnswerHub more frequently. 37% (110) of Unity users and 31% (60) of UE4 users reported having accessed the communities regularly. These percentages rank third when compared to the other resources available for the engines.

Respondents who did not access the Q&A communities regularly also expressed their preference for other resources. When asked the reason for not accessing communities more frequently, 21 of the 49 respondents for Unity Answers and 5 of the 16 respondents for the UE4 AnswerHub mentioned using other means to find the solution to their problems.

Over half of the respondents did not actively participate in the Q&A communities. Only 40% (83) of respondents who accessed Unity Answers and 52% (51) who accessed the UE4 AnswerHub actively interacted with the communities by creating posts. Meanwhile, over 90% of respondents (191 on Unity Answers and 93 on the UE4 AnswerHub) used the communities to search for questions related to their problems.

Most of the respondents who reported having posted on the communities asked questions (71 on Unity Answers and 44 on the UE4 AnswerHub), while only a few gave answers (33 and 22). These results are similar to those found for Stack Overflow, where most users ask rather than answer questions [73]. On the UE4 AnswerHub, one in every five respondents (20%) also used the community to report bugs.

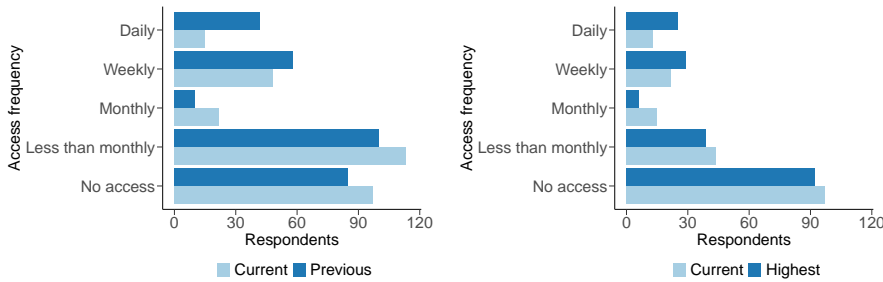
Respondents have a mostly negative view of the communities. When asked about their likes and dislikes about the communities, 52 (57%) of the 92 that answered the question expressed disapproval for Unity Answers and 23 (64%) of the 36 for the UE4 AnswerHub. In contrast, only 33 (36%) respondents indicated some approval for Unity Answers and 10 (28%) for the

UE4 AnswerHub. Other responses were not related to the communities or did not convey a definite opinion.

For Unity Answers, 21 (23%) respondents complained about the low quality of answers. Eight (9%) of those respondents emphasized that many answers are obsolete and have broken links to other websites, which is also a problem faced by Stack Overflow [81]. Six (17%) respondents also mentioned the low quality of answers on the UE4 AnswerHub. Additionally, respondents criticized both communities for having unanswered questions (17 of the total 128 responses, 13%), for the overall website design and organization (seven responses, 5%), and the communities' low activity (11 responses, 9%). Six (5%) respondents mentioned the existence of resources other than Unity Answers for finding the answers they needed.

In contrast, thirteen of the 128 responses (10%) showed approval for the communities as they served their purpose of finding solutions. Other responses did not highlight the Q&A websites themselves but focused on the members of the community. Fifteen (12%) responses praised the communities, for being large, supportive, active, and experienced.

The frequency with which respondents accessed the communities decreased. Fig. 9(a) and Fig. 9(b) show the distribution of the previous and current access frequencies reported by respondents who accessed the Q&A websites. We observed a decrease in the “daily” and “weekly” categories of access frequency in both communities, while the other categories increased. We found that over half of the respondents that accessed the communities regularly (65 of 110 for Unity Answers and 33 of 60 for the UE4 AnswerHub) either decreased or stopped their access to the communities.



(a) Respondents' access frequencies of Unity Answers (b) Respondents' access frequencies of the UE4 AnswerHub

Fig. 9 Current and previous frequencies with which survey respondents accessed the studied Q&A communities.

The main reason respondents gave for decreasing their access was becoming more experienced and not needing the communities' help anymore. Some respondents also mentioned using other resources to find the answers they needed, while others reduced or stopped their use of the game engines. The

respondents who stopped accessing the communities altogether reported similar motives as those that decreased their access. The only reason respondents gave for increasing their access frequency was the still existent necessity of answering their questions and acquiring more knowledge.

Summary: Despite knowing of the Q&A communities' existence, most survey respondents would rather use other learning resources. Of the ones that used the communities, most did not show support for them and did not post any questions or answers. The respondents who regularly accessed the communities decreased their access frequency as they acquired knowledge.

5 Implications of our findings

Our findings indicate that there are several challenges in maintaining healthy Q&A communities, especially when they become more popular and cater to a specific niche. All of the studied communities showed some sign of decline, despite those changes being more subtle in the smaller ones.

As the communities grew, new and inexperienced users joined and increased the number of questions. At the same time, experienced users that could provide solutions to those questions became inactive for possibly feeling overloaded. More questions went unanswered and unresolved, causing frustration for newcomers. Users started seeking other resources for solving their problems, and community growth halted or even decreased. These communities could not yet recover after experiencing this decline.

Meanwhile, we have shown that the studied Q&A websites have different characteristics, discussing topics in different proportions and with varying levels of complexity of abstraction. These findings help in guiding game developers towards the websites that are most effective in answering the type of question they have. To further aid game developers in choosing the right Q&A website to post their questions, we created a flowchart (Figure 10) detailing our recommended decision process.

We based the flowchart on our analysis of the topics that each community is most effective in discussing. For example, we recommend that game developers post questions related to bugs and errors in Unity Answers and the UE4 AnswerHub given that those two websites had several posts related to these topics. Similarly, we recommend that game developers post programming questions on Stack Overflow or the Game Development Stack Exchange, as those topics were some of the most discussed in the two websites. We also direct game developers to the Game Development Stack Exchange if their questions fit a set of five topics (Rendering, Geometry, Movement, Game mechanics, and Positioning) that have been more effectively handled by that website.

By directing game development questions to the websites that are more likely to welcome and solve them, we can help the studied Q&A websites in mitigating their decline. For example, reducing the amount of questions asked in the largest websites and directing them to the smaller ones, we may avoid

overloading a single community with a flood of questions. Moreover, we can increase the communities' effectiveness by reducing the number of questions that go unanswered and unresolved.

However, we note that many uncontrollable factors may be at play when it comes to what is causing the communities' decline and Q&A website administrators should also act towards improving the websites and providing a better environment for nurturing a community. While we cannot infer causality and provide fail-proof advice, our findings also hint at actions that can help in lifting these communities, such as maintaining an active staff of moderators and investing in gamification features to motivate users' participation [8, 19, 28, 67, 74, 84].

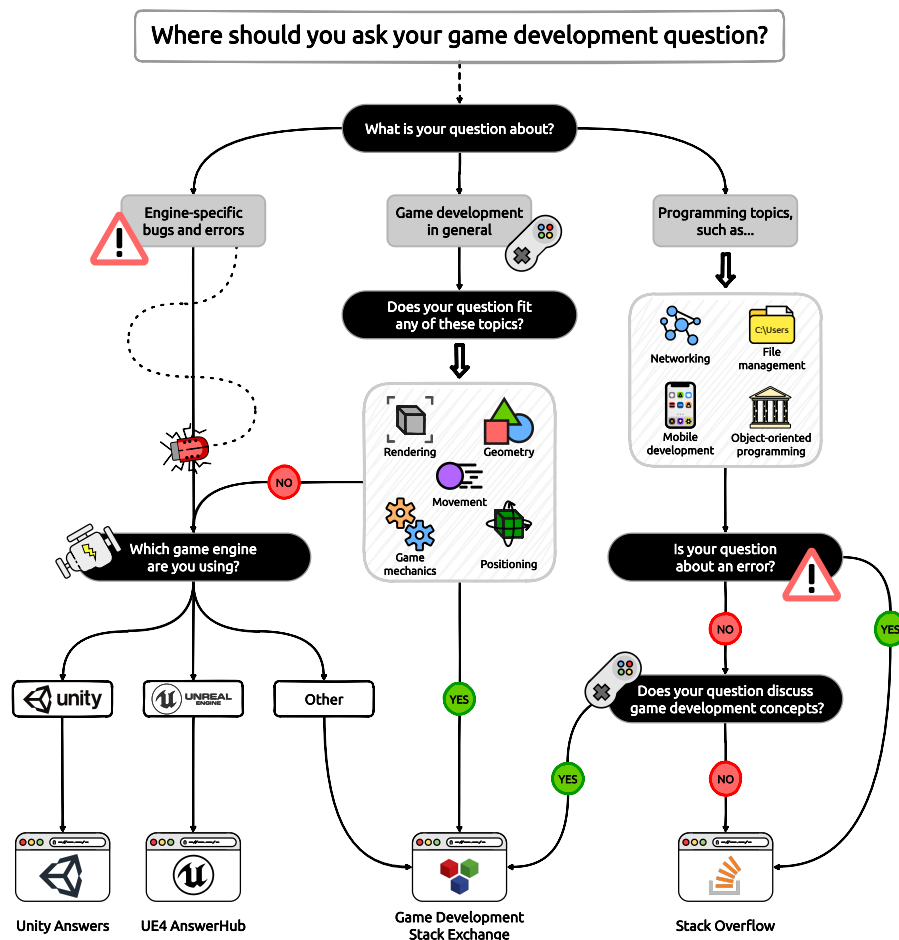


Fig. 10 Flowchart showing our recommended decision process to choosing in which Q&A website to post questions about game development.

6 Threats to validity

In this section we discuss the threats to the internal and external validities of our study.

6.1 Construct validity

The choices we made in our methodology (Section 3) may have directly affected our findings. We have identified game development questions on Stack Overflow using a group of manually-selected tags. Therefore, we may have missed other game development questions with different tags than the ones we selected.

The LDA algorithm can provide different results based on its training input, and the preprocessing steps we perform on the data can have a direct impact in its output. Moreover, the output also depends on the initial parameters, and altering them might change the resulting set of topics.

We have experimented using additional preprocessing steps (such as pruning the corpora) and other sets of parameters for modeling topics. The results of these experiments were similar to the ones we presented, with slight variations in the meaning of topics and words assigned to them. Ultimately, we chose those that offered the best set of topics with our minimum possible interference.

Furthermore, we based our choice of the labels and categories assigned to each topic on our own experience and expertise with the game and software development disciplines. While we cannot guarantee that these labels precisely describe all of the subjects discussed in each topic, we bolstered our decision of labels with a manual analysis of the content of posts. Therefore, we are confident these labels can approximately describe their topics.

6.2 Internal validity

While we have not drawn any conclusions related to causality in our data, we have indicated measures that may help developers and communities based on our findings (Section 5). However, we note that many unmeasured factors may have an effect on the patterns we observed. Thus, we cannot guarantee that the measures we recommended will have an effect on these websites and their communities. Yet, we still believe our findings are useful for game developers and game development communities by bringing more understanding of their past discussions.

The proposed taxonomy we used for classifying the topics obtained by the LDA algorithm was not validated by a video game expert. Future studies should validate the taxonomy in a large-scale study with a diverse group of game developers who have broad experiences with developing games with the studied game engines and some familiarity with the game development topics and discussions.

6.3 External validity

In this paper, we have only analyzed data from three game development Q&A websites and questions related to game development on Stack Overflow. We note, however, that those websites are the largest of their kind, and we believe that they could reflect the trends in the game development community as a whole. Yet, our findings may be limited to the developer communities that interact with them. Future studies should investigate how our findings apply to other groups of developers that use other websites.

Furthermore, our findings may only apply to Q&A websites that discuss game development, and not to other types of websites or general-purpose Q&A websites. However, our analysis focuses on features common to many other Q&A websites and may serve as a foresight of what other communities experience if they meet similar conditions. Further studies should compare our findings to other Q&A communities.

Part of our study focused on analyzing specific aspects of the Unity engine and the Unreal Engine 4. While the majority of game developers use these two engines, future studies need to investigate how our findings hold for other game engines.

Our survey's responses come from game developers that were active members of the selected game development communities when we shared the survey. Also, we only asked questions about Unity Answers and the UE4 AnswerHub, and our findings may not apply to other Q&A websites for game developers. Thus, this specific sample may not be an accurate representation of the whole game development community. Nevertheless, these respondents are still community members and their responses provide valuable information on the part of those communities they represent.

7 Conclusion

In this study, we analyzed game development Q&A communities using data from Unity Answers, the UE4 AnswerHub, the Game Development Stack Exchange, and game development-related questions from Stack Overflow. We also analyzed 347 responses game developers gave to our survey. We have explored how the communities evolved using measures of their activity and effectiveness, while also identifying what factors correlate to these measures' changes. We explored the topics discussed in those communities, analyzing these topics' distribution and identifying the most frequently discussed topics. We compared the topics among themselves and between the four websites using characteristics extracted from their posts. Finally, we examined how survey respondents used these communities through their access frequencies and their contributions to the websites. We also compared the respondents' usage of the websites to other online learning resources. Our most important findings are:

- (1) The Q&A communities are declining, with their activity levels and effectiveness decreasing over time.

(2) Experienced users contribute to the community by answering and resolving a large number of questions. These users stopped using the community, reducing the communities' effectiveness.

(3) Game developers mainly discussed topics related to game development, with topics about Bug Reports having accrued a larger number of posts on Unity Answers and the UE4 AnswerHub.

(4) Topics have shown varying levels of abstraction and complexity, differing in terms of the length of their posts and the frequency with which code is discussed in them.

(5) Users do not often participate by posting questions or answers, have a mostly negative view of the community, and have reduced their access frequencies.

These findings bring new understanding about game developers, their communities, and the topics that they discuss. Based on these findings, we provide recommendations that help game developers in directing their questions to communities that can handle them better, which can aid the communities to overcome their decline and become more effective in the future.

References

1. About Stack Overflow. URL <https://stackoverflow.com/company>. Accessed: August 14, 2020
2. Stack Overflow's 2020 developer survey. URL <https://insights.stackoverflow.com/survey/2020>. Accessed: August 14, 2020
3. Adamic, L.A., Zhang, J., Bakshy, E., Ackerman, M.S.: Knowledge sharing and yahoo answers: everyone knows something. In: Proceedings of the 17th international conference on World Wide Web, pp. 665–674 (2008)
4. Ahmad, A., Feng, C., Ge, S., Yousif, A.: A survey on mining Stack Overflow: Question and answering (Q&A) community. *Data Technologies and Applications* (2018)
5. Ahmed, S., Bagherzadeh, M.: What do concurrency developers ask about? A large-scale study using Stack Overflow. In: Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, pp. 1–10 (2018)
6. Allamanis, M., Sutton, C.: Why, when, and what: analyzing Stack Overflow questions by topic, type, and code. In: 2013 10th Working Conference on Mining Software Repositories (MSR), pp. 53–56. IEEE (2013)
7. An, L., Mlouki, O., Khomh, F., Antoniol, G.: Stack Overflow: A code laundering platform? In: 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER), pp. 283–293. IEEE (2017)
8. Anderson, A., Huttenlocher, D., Kleinberg, J., Leskovec, J.: Discovering value from community activity on focused question answering sites: A case study of Stack Overflow. In: Proceedings of the 18th ACM SIGKDD

- international conference on Knowledge discovery and data mining, pp. 850–858 (2012)
9. Asaduzzaman, M., Mashiyat, A.S., Roy, C.K., Schneider, K.A.: Answering questions about unanswered questions of Stack Overflow. In: 2013 10th Working Conference on Mining Software Repositories (MSR), pp. 97–100. IEEE (2013)
 10. Bajaj, K., Pattabiraman, K., Mesbah, A.: Mining questions asked by web developers. In: Proceedings of the 11th Working Conference on Mining Software Repositories, pp. 112–121 (2014)
 11. Bangash, A.A., Sahar, H., Chowdhury, S., Wong, A.W., Hindle, A., Ali, K.: What do developers know about machine learning: a study of ml discussions on stackoverflow. In: 2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR), pp. 260–264. IEEE (2019)
 12. Barua, A., Thomas, S.W., Hassan, A.E.: What are developers talking about? An analysis of topics and trends in Stack Overflow. *Empirical Software Engineering* **19**(3), 619–654 (2014)
 13. Beyer, S., Macho, C., Di Penta, M., Pinzger, M.: What kind of questions do developers ask on Stack Overflow? A comparison of automated approaches to classify posts into question categories. *Empirical Software Engineering* **25**(3), 2258–2301 (2020)
 14. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *Journal of machine Learning research* **3**(Jan), 993–1022 (2003)
 15. Blodgett, B.M., Salter, A.: Hearing ‘lady game creators’ tweet: #1reasonwhy, women and online discourse in the game development community. In: 14th Annual Conference for the Association of Internet Researchers (AoIR) (2013)
 16. Blodgett, B.M., Salter, A.: # 1reasonwhy: Game communities and the invisible woman. In: *Foundations of Digital Games* (2014)
 17. Brookes, J., Warburton, M., Alghadier, M., Mon-Williams, M., Mushtaq, F.: Studying human behavior with virtual reality: The Unity experiment framework. *Behavior research methods* pp. 1–9 (2019)
 18. Budhiraja, A., Reddy, R., Shrivastava, M.: Lwe: Lda refined word embeddings for duplicate bug report detection. In: Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings, pp. 165–166 (2018)
 19. Cavusoglu, H., Li, Z., Huang, K.W.: Can gamification motivate voluntary contributions? The case of StackOverflow Q&A community. In: Proceedings of the 18th ACM conference companion on computer supported cooperative work & social computing, pp. 171–174 (2015)
 20. Chen, L., Baird, A., Straub, D.: Why do participants continue to contribute? Evaluation of usefulness voting and commenting ootivational affordances within an online knowledge community. *Decision Support Systems* **118**, 21–32 (2019)
 21. Chen, T.H., Thomas, S.W., Hassan, A.E.: A survey on the use of topic models when mining software repositories. *Empirical Software Engineering* **21**(5), 1843–1919 (2016)

22. Christopoulou, E., Xinogalos, S.: Overview and comparative analysis of game engines for desktop and mobile devices. *International Journal of Serious Games* **4**(4) (2017)
23. Cliff, N.: Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological bulletin* **114**(3), 494 (1993)
24. Cowan, B., Kapralos, B.: A survey of frameworks and game engines for serious game development. In: 2014 IEEE 14th International Conference on Advanced Learning Technologies, pp. 662–664. IEEE (2014)
25. Cox, D.R., Stuart, A.: Some quick sign tests for trend in location and dispersion. *Biometrika* **42**(1/2), 80–95 (1955)
26. Cristie, V., Berger, M.: Game engines for urban exploration: Bridging science narrative for broader participants. In: *Playable Cities*, pp. 87–107. Springer (2017)
27. Cummaudo, A., Vasa, R., Barnett, S., Grundy, J., Abdelrazek, M.: Interpreting cloud computer vision pain-points: A mining study of Stack Overflow. *arXiv preprint arXiv:2001.10130* (2020)
28. Deng, S., Tong, J., Lin, Y., Li, H., Liu, Y.: Motivating scholars’ responses in academic social networking sites: An empirical study on ResearchGate Q&A behavior. *Information Processing & Management* **56**(6), 102082 (2019)
29. Epp, R., Lin, D., Bezemer, C.P.: An empirical study of trends of popular virtual reality games and their complaints. *IEEE Transactions on Games* pp. 1–12 (2021)
30. Fang, C., Zhang, J.: Users’ continued participation behavior in social Q&A communities: A motivation perspective. *Computers in Human Behavior* **92**, 87–109 (2019)
31. Fischer, F., Böttinger, K., Xiao, H., Stransky, C., Acar, Y., Backes, M., Fahl, S.: Stack Overflow considered harmful? The impact of copy&paste on android application security. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 121–136. IEEE (2017)
32. Fisher, S.J., Harvey, A.: Intervention for inclusivity: Gender politics and indie game development. *Loading...* **7**(11) (2013)
33. Ford, D., Lustig, K., Banks, J., Parnin, C.: “we don’t do that here” how collaborative editing with mentors improves engagement in social Q&A communities. In: *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–12 (2018)
34. Foxman, M.: United we stand: Platforms, tools and innovation with the Unity game engine. *Social Media+ Society* **5**(4), 2056305119880177 (2019)
35. Fu, H., Oh, S.: Quality assessment of answers with user-identified criteria and data-driven features in social Q&A. *Information Processing & Management* **56**(1), 14–28 (2019)
36. Guan, T., Wang, L., Jin, J., Song, X.: Knowledge contribution behavior in online Q&A communities: An empirical investigation. *Computers in Human Behavior* **81**, 137–147 (2018)
37. Han, J., Shihab, E., Wan, Z., Deng, S., Xia, X.: What do programmers discuss about deep learning frameworks. *Empirical Software Engineering*

- 25**(4), 2694–2747 (2020)
38. Harper, F.M., Raban, D., Rafaeli, S., Konstan, J.A.: Predictors of answer quality in online q&a sites. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 865–874 (2008)
 39. Harrap, R., Hutchinson, D., Sala, Z., Ondercin, M., DiFrancesco, P.: Our GIS is a game engine: Bringing Unity to spatial simulation of rockfalls. In: *GeoComputation* (2019)
 40. Harvey, A.: Becoming gamesworkers: Diversity, higher education, and the future of the game industry. *Television & New Media* **20**(8), 756–766 (2019)
 41. Hong, Z., Deng, Z., Evans, R., Wu, H.: Patient questions and physician responses in a Chinese health Q&A website: Content analysis. *Journal of Medical Internet Research* **22**(4), e13071 (2020)
 42. Hu, H., Wang, S., Bezemer, C.P., Hassan, A.E.: Studying the consistency of star ratings and reviews of popular free hybrid android and ios apps. *Empirical Software Engineering* **24**(1), 7–32 (2019)
 43. Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., Zhao, L.: Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications* **78**(11), 15169–15211 (2019)
 44. Jin, J., Li, Y., Zhong, X., Zhai, L.: Why users contribute knowledge to online communities: An empirical study of an online social Q&A community. *Information & management* **52**(7), 840–849 (2015)
 45. Kanode, C.M., Haddad, H.M.: Software engineering challenges in game development. In: *2009 Sixth International Conference on Information Technology: New Generations*, pp. 260–265. IEEE (2009)
 46. Koksall, I.: Video gaming industry & its revenue shift (2019). URL <https://www.forbes.com/sites/ilkerkoksal/2019/11/08/video-gaming-industry--its-revenue-shift/#12d74894663e>. Accessed: August 14, 2020
 47. Lin, B., Zampetti, F., Bavota, G., Di Penta, M., Lanza, M.: Pattern-based mining of opinions in Q&A websites. In: *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pp. 548–559. IEEE (2019)
 48. Linares-Vásquez, M., Bavota, G., Di Penta, M., Oliveto, R., Poshyvanyk, D.: How do API changes trigger Stack Overflow discussions? A study on the Android SDK. In: *proceedings of the 22nd International Conference on Program Comprehension*, pp. 83–94 (2014)
 49. Linares-Vásquez, M., Dit, B., Poshyvanyk, D.: An exploratory analysis of mobile development issues using stack overflow. In: *2013 10th Working Conference on Mining Software Repositories (MSR)*, pp. 93–96. IEEE (2013)
 50. Lukins, S.K., Kraft, N.A., Etzkorn, L.H.: Bug localization using latent dirichlet allocation. *Information and Software Technology* **52**(9), 972–990 (2010)
 51. Mamykina, L., Manóim, B., Mittal, M., Hripcsak, G., Hartmann, B.: Design lessons from the fastest Q&A site in the west. In: *Proceedings of the*

- SIGCHI conference on Human factors in computing systems, pp. 2857–2866 (2011)
52. Mann, H.B., Whitney, D.R.: On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics* pp. 50–60 (1947)
 53. Messaoudi, F., Ksentini, A., Simon, G., Bertin, P.: Performance analysis of game engines on mobile and fixed devices. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* **13**(4), 1–28 (2017)
 54. Morris, M.R., Teevan, J., Panovich, K.: What do people ask their social networks, and why? a survey study of status message q&a behavior. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 1739–1748 (2010)
 55. Movshovitz-Attias, D., Movshovitz-Attias, Y., Steenkiste, P., Faloutsos, C.: Analysis of the reputation system and user contributions on a question answering website: StackOverflow. In: *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)*, pp. 886–893. IEEE (2013)
 56. Nadi, S., Treude, C.: Essential sentences for navigating Stack Overflow answers. In: *2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pp. 229–239. IEEE (2020)
 57. Parnin, C., Treude, C., Grammel, L., Storey, M.A.: Crowd documentation: Exploring the coverage and the dynamics of API discussions on Stack Overflow. *Georgia Institute of Technology, Tech. Rep* **11** (2012)
 58. Pascarella, L., Palomba, F., Di Penta, M., Bacchelli, A.: How is video game development different from software development in open source? In: *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*, pp. 392–402. IEEE (2018)
 59. Porter, M.F., et al.: An algorithm for suffix stripping. *Program* **14**(3), 130–137 (1980)
 60. Procaci, T.B., Nunes, B.P., Nurmikko-Fuller, T., Siqueira, S.W.: Finding topical experts in question & answer communities. In: *2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT)*, pp. 407–411. IEEE (2016)
 61. Procaci, T.B., Siqueira, S.W., Nunes, B.P., Nurmikko-Fuller, T.: Modelling experts behaviour in Q&A communities to predict worthy discussions. In: *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, pp. 291–295. IEEE (2017)
 62. Ravi, S., Pang, B., Rastogi, V., Kumar, R.: Great question! question quality in community q&a. In: *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8 (2014)
 63. Ray, B., Posnett, D., Filkov, V., Devanbu, P.: A large scale study of programming languages and code quality in github. In: *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 155–165 (2014)

64. Romano, J., Kromrey, J.D., Coraggio, J., Skowronek, J., Devine, L.: Exploring methods for evaluating group differences on the nsse and other surveys: Are the t-test and cohen's d indices the most appropriate choices. In: annual meeting of the Southern Association for Institutional Research, pp. 1–51. Citeseer (2006)
65. Rosen, C., Shihab, E.: What are mobile developers asking about? A large scale study using Stack Overflow. *Empirical Software Engineering* **21**(3), 1192–1223 (2016)
66. Sadowski, C., Stolee, K.T., Elbaum, S.: How developers search for code: A case study. In: Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering, pp. 191–201 (2015)
67. Santos, T., Burghardt, K., Lerman, K., Helic, D.: Can badges foster a more welcoming culture on Q&A boards? In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 14, pp. 969–973 (2020)
68. Shen, X., Jia, A.L., Shen, S., Dou, Y.: Helping the ineloquent farmers: Finding experts for questions with limited text in agricultural Q&A communities. *IEEE Access* **8**, 62238–62247 (2020)
69. Somasundaram, K., Murphy, G.C.: Automatic categorization of bug reports using latent dirichlet allocation. In: Proceedings of the 5th India software engineering conference, pp. 125–130 (2012)
70. Treude, C., Barzilay, O., Storey, M.A.: How do programmers ask and answer questions on the web? (NIER track). In: Proceedings of the 33rd international conference on software engineering, pp. 804–807 (2011)
71. Treude, C., Robillard, M.P.: Augmenting API documentation with insights from Stack Overflow. In: 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE), pp. 392–403. IEEE (2016)
72. Wang, S., Chen, T.H., Hassan, A.E.: Understanding the factors for fast answers in technical Q&A websites. *Empirical Software Engineering* **23**(3), 1552–1593 (2018)
73. Wang, S., Lo, D., Jiang, L.: An empirical study on developer interactions in StackOverflow. In: Proceedings of the 28th Annual ACM Symposium on Applied Computing, pp. 1019–1024 (2013)
74. Wang, Y.: The price of being polite: politeness, social status, and their joint impacts on community Q&A efficiency. *Journal of Computational Social Science* pp. 1–22 (2020)
75. Whitson, J.R.: Voodoo software and boundary objects in game development: How developers collaborate and conflict with game engines and art tools. *new media & society* **20**(7), 2315–2332 (2018)
76. Wilcoxon, F.: Individual comparisons by ranking methods. In: Breakthroughs in statistics, pp. 196–202. Springer (1992)
77. Wu, Y., Wang, S., Bezemer, C.P., Inoue, K.: How do developers utilize source code from Stack Overflow? *Empirical Software Engineering* **24**(2), 637–673 (2019)
78. Yang, X.L., Lo, D., Xia, X., Wan, Z.Y., Sun, J.L.: What security questions do developers ask? a large-scale study of stack overflow posts. *Journal of Computer Science and Technology* **31**(5), 910–924 (2016)

79. Zahedi, M., Rajapakse, R.N., Babar, M.A.: Mining questions asked about continuous software engineering: A case study of stack overflow. In: *Proceedings of the Evaluation and Assessment in Software Engineering*, pp. 41–50 (2020)
80. Zhang, H., Wang, S., Chen, T.H., Hassan, A.E.: Reading answers on stack overflow: Not enough! *IEEE Transactions on Software Engineering* (2019)
81. Zhang, H., Wang, S., Chen, T.H.P., Zou, Y., Hassan, A.E.: An empirical study of obsolete answers on Stack Overflow. *IEEE Transactions on Software Engineering* (2019)
82. Zhang, X., Liu, S., Chen, X., et al.: Social capital, motivations, and knowledge sharing intention in health Q&A communities. *Management Decision* (2017)
83. Zhang, Y., Lu, T., Phang, C.W., Zhang, C.: Scientific knowledge communication in online Q&A communities: Linguistic devices as a tool to increase the popularity and perceived professionalism of knowledge contribution. *Journal of the Association for Information Systems* **20**(8), 3 (2019)
84. Zhou, J., Wang, S., Bezemer, C.P., Hassan, A.E.: Bounties on technical Q&A sites: a case study of Stack Overflow bounties. *Empirical Software Engineering* **25**(1), 139–177 (2020)