How Do Bot Developers Perceive Bot Development? A Survey

André M. Pinheiro, Caio S. Rabello, Leonardo B. Furtado,
Gustavo Pinto, Cleidson R. B. de Souza

[1]Universidade Federal do Pará (UFPA)
Caixa Postal 66.075-110 – Belém – PA – Brazil

{andremirandap93,xcaiosr,srleonardofurtado}@gmail.com,

gpinto@ufpa.br,cleidson.desouza@acm.org

Abstract. *Software bots are becoming more and more popular, this is due to the fact that they are a tool that can be used in different contexts. With this, software developer has more interest in developing bots, however they may have to face challenges intrinsic to the development of bot software. With this, we seek to understand the profile of bot developers, what motivates them, or what challenges they face when dealing with bot development. To shed an initial light on this direction,we conducted a survey with 43 Github users who have been involved (showingtheir interest or actively contributing to) in bot software projects.*

1. Introduction

Software bots date back to the 1950s when Alan Turing proposed the *imitation game* in which a machine gave the impression of being a human being, i.e., that a machine could think and act like a human [Wessel et al. 2018, Turing 1950]. Recently, software bots (or bots, for simplicity) are becoming increasingly popular among users of instant messaging due to the possibility of integrating secondary services to these applications through them [Klopfenstein et al. 2017, Lebeuf et al. 2018]. This includes from simple services such as ordering a meal, booking a restaurant or asking for a taxi from an application that (s)he knows and trusts [Klopfenstein et al. 2017] to more complex functionalities like a GitHub integration to Slack, for example[1].

Thanks to the rise of social coding websites (e.g., GitHub or GitLab) and the increasing complexity of the software development process with new APIs, software developers have to integrate their development activities into a complex workflow. As a consequence, one single commit has the potential of triggering a chain of events in several distinct services (e.g., running a testing suite locally, starting a continuous integration and/or delivery service(s) in the company infrastructure, deploying a fresh version to the cloud, etc). In this complex landscape, in which errors can (and will) happen in one or many unknown places, bots arise as a suitable solution to assist developers in producing documentation, software testing, communicating with users and among developers, code deployment, or addressing slow feedback loops between developers, etc. [Storey and Zagalsky 2016a].

Not only important for the software development practice, bots are useful for several other industries in which customer service play a vital role. To answer the far from

[1] https://github.com/integrations/slack

trivial questions asked by impatient customers, bots became intrinsically sophisticated software systems. With the widespread use of bots, one could expect that such complexity will skyrocket in very few years. In particular, Gartner is predicting that 25% of every costumer service operation will rely on bots by 2020[2]. As a consequence, bot software developers have to master processes, techniques, and tools that are otherwise not readily available items of their programming arsenal.

Although some preliminary work focused on the use of bots in software development projects [Lin et al. 2016, Pérez-Soler et al. 2017, Wessel et al. 2018, Lebeuf et al. 2017, Bradley et al. 2018] , little is known about bot developers, what motivate them, or what challenges they face. This work is an additional step to understand the bot developer [Storey and Zagalsky 2016b]. Since this research is still in its early stages, in this paper we focus on high-level exploratory research questions. More concretely, the questions we are trying to answer are:

- **RQ1:** What is the profile of bot developers?
- **RQ2:** How do bot developers work?
- **RQ3:** What challenges bot developers face when developing bots?
- **RQ4:** What motivated developers to develop his(ers) most recent bot?

To answer these questions, we deployed a questionnaire to 2,306 GitHub users that contributed or at least had shown interest in bot software projects. We collected 43 answers of bot developers. Through a quantitative and qualitative analysis of the survey responses, our results suggest that most of the respondents have more software developing experience than bot developing experience, which is expected. The respondents also reported having created a small number of bots, which mostly have been used by less than 100 people. Bot developers work mainly by themselves, while development teams were composed mostly by a maximum of 5 members. More interestingly, a considerable amount of the respondents confirmed having difficulties implementing a specific bot feature, usually something related to machine learning. Other reported problems including integrating with other tools, tools usage, conducting tests, and finding information about bot development. As regarding the motivations, we found that personal needs (e.g., automating the own house) and workplace assistance (e.g., to facilitate communication with clients) play a major role.

2. Method

2.1. Data Collection

To answer our research questions, our initial idea was to conduct interviews, but we had difficulties in finding available bot developers. Therefore, we decided to collect data using a questionnaire. Our questionnaire had 21 questions with 15 open and 6 closed questions. The questionnaire had three main sections. In the first section we asked demographic questions including the respondent's personal experience with software development and bot development. The second section was interested in the latest bot developed by the respondent. Finally, the third section addressed technical activities employed in the process of developing this specific bot. Some of questions covered in the survey include:

[2] https://www.gartner.com/en/newsroom/press-releases/2018-02-19-gartner-says-25-percent-of-customer-service-operations-will-use-virtual-customer-assistants-by-2020

If you have developed bots, how long have you been developing them? Why did you start developing bots? What was your motivation? What are the difficulties or challenges of developing bots? Why did you develop this specific bot? What motivated you for developing this bot? If this bot is an integration, what applications, tools, networks, etc, does it integrate with? If you developed this bot as a team, how many other people are involved in this process, including you? The full questionnaire is available online at `https://goo.gl/forms/f59U1YPF519XaUml1` for replication purposes.

The questionnaire was sent to software developers that have contributed to a GitHub project that have tagged itself under the ”slack bot” topic. By contributing, we mean of forking, staring, or providing a pull-request to the project. To find these projects we used the search bar available at the GitHub main page. Although this was the method we used to search projects, this does not mean that only slack bot developers would receive the questionnaire since most of the emails we sent were addressed to users that contributed to bot development frameworks for different platforms including slack. See Table 1 for more details on the selected GitHub projects. To avoid selecting unmaintained projects, we filter out projects that did not have any activity (i.e. commits) during the period of one year. We also did *not* select bots that were too specific, e.g., a bot that auto-likes someone picture in instagram. Consequently, we obtained 10 unique open source projects with these characteristics. By unique we mean that none of the projects m is a fork from each other. The list of all projects is presented on Table 1. It is possible to notice that there are 6 bot development frameworks, i.e., these projects aim to facilitate the development of bots. The other projects were bots. When studying the domain of these bots, we found that they are greatly diverse, covering areas such as frameworks for bot development, and integration of social media/coding networks to Slack.

Table 1. Bots characteristics

Repository	Type of project	# of Stars	# of Forks	# of Commits	Last Commit	Language
howdyai/botkit	Framework	8,717	1,922	2,392	01-31-2019	JavaScript
slackapi/hubot-slack	Framework	2,103	608	752	01-25-2019	CoffeeScript
slackapi/python-rtmbot	Framework	618	270	175	03-30-2018	Python
Rampatra/jbot	Framework	960	251	284	01-23-2019	Java
Netflix/hubcommander	Bot	1,103	139	161	10-14-2018	Python
mishk0/slack-bot-api	Framework	1,047	115	143	10-17-2018	JavaScript
integrations/slack	Bot	640	115	2,081	01-29-2019	JavaScript
looker/lookerbot	Bot	91	96	368	08-29-2018	TypeScript
shomali11/slacker	Framework	219	29	93	01-28-2019	Go
Weetbix/facebot	Bot	80	21	123	07-11-2018	Javascript

We found a total of 19.144 users that have forked or starred a project, but only 2,165 have emails publicly available. For the 1,409 PRs distributed by the project repositories we selected we found a total of 216 users that have emails publicly available. We took the precaution not to send the email more than once to the same developer.

An invitation to answer this questionnaire was sent to a total of 2,306 software developers[3]. We sent the questionnaire in two batches. The first batch target software developers who forked or stared the project (comprising 403 emails of users that forked a project and 1,854 emails of users that only stared a project, totaling 2,257 emails, though

[3] The first set of 400 emails were sent as CC, by mistake. We believe that by some developers might have not answered the survey because their email was exposed to other developers.

92 emails could not be sent due to technical issues). We sent these emails during the period of December 3rd to December 6th. Then, we sent a reminder on December 12nd, 2018. The second batch of email invitations was aimed to developers who performed pull-requests to the same set of projects and who did not have received an email during the first period. This resulted in 141 invitations to software developers of the 216 developers we found. We sent the email invitations on January 14th, 2019, and a reminder was sent on January 21st. Overall, we received a total of 47 answers. However, we discarded four of them because the respondents of these answers did not have previous bot development experience. At the end of this process, we ended up with 43 responses to our questionnaire. Some thoughts about this low response rate are discussed at Section 5.

2.2. Data Analysis

Closed questions were analyzed using bars and distribution graphics provided by Google Docs. We adopted what grounded theory calls open-coding [Strauss and Corbin 1990] to analyzed the open questions. The first and third author read all the answers and derived themes to classify them answers on the motivations and difficulties in developing bots. These themes were reviewed for consistency by the second author. No inconsistencies were identified.

3. Results

We present our results according to the research questions. We highlight the main themes that emerged along with quotes from the open questions. The developers who participated in our study are identified as D1 – D43.

3.1. RQ1: What is the profile of bot developers?

Our questionnaire was answered by software developers that live mainly in Europe (30.2%) or the United States (27.9%). The rest of respondents live in Asia (21%), South America (18,6%), or Africa (2.3%). When analyzing the responses of the developers, it can be seen that the majority of the respondents (79%) have more than five years of software development experience. However, we noted that about 70% of the respondents have up to two years of experience developing bots. This short experience with bot development might also justify the fact that our respondents had developed a small amount of bots (62% of the respondents have developed up to 3 bots). However, we also noted that there are very prolific members in our group of respondents: 9.3% of them have developed more than 10 bots. It is also worthy noticing that 2.3% of our survey respondents reported developing bots for over 10 years. We plan to interview these developers to find out *what* exactly they mean by bots and whether their understanding of bots is the same as the other respondents.

Finally, as regarding *what motivated developers to start developing bots*, 41% of the respondents started to develop bots to automate tasks (e.g., “Simplify the task and reduce human efforts”, “Automate a request response flow”). However, we also noticed other motivations, including the purpose of learning a new technology (25%) (e.g., “I was interested in natural language processing”, “For Class”), commercial or financial purposes (e.g., “For a customer requirement”, “financial gain”), or even for entertainment (16%) (e.g., “For fun and convenience”, “Entertainment purpose for private chat group”).

3.2. RQ2: How do bot developers work?

In this research question we investigate collaborative aspects of bot development. First, we observed that the majority of the respondents (65%) worked alone when developing the bot. Moreover, 35% of bot developers work collaborativel. For those that work in groups, 40% of them work with groups of two. Interestingly, 13.3% of the respondents reported to work on groups with more than 6 developers. Moreover, the majority of the groups have developers working collocated (61.9%), although distributed bot development is fairly common (38.1%). To aid bot development, these distributed teams use well-known communication and collaboration tools, such as Slack, Github, Gitlab, Jira, and VScode.

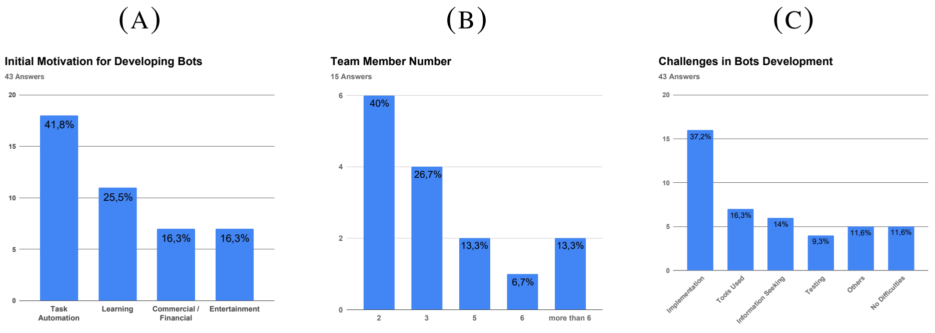

Figure 1. (a) means Initial motivation for developing bots, (b) means number of team members, and (c) means challenges in Bots Development.

Finally, as mentioned in Section 2.1, the second part of our survey was about the latest bot developed by the respondent. Specifically, we asked whether the latest developed bot was intended to be used by one person or a group of people, i.e., whether the bot was collaborative or not. We observed that 20.9% of the bots should be used by one person, 41.9% of bots should be used by a group of people, and 37.2% of responses that it can be used by either only one person or a group. This finding poses several challenges. For instance, since developers work mostly alone to develop their bots (65% of them), they might have a hard time trying to test bot features that are intended to work in a group. We will return to this issue on the Discussion section.

3.3. RQ3: What challenges bot developers face when developing bots?

The difficulties reported by the developers are mainly related to the tools used to develop bots, bot implementation, testing and finding information about bot development. Figure 1-(c) summarizes the results, while the following paragraphs explain each one of these categories.

Bot implementation: According to the survey respondents, bot developers should have a more extensive development arsenal to deal with, for instance, how to respond to users actions, which are intrinsically dynamic. To illustrate, D5 mentioned that "*The mapping of user intent and the response to give back [is a challenge]. Currently, [I have] been using the AIML to develop bots aided with the logic out of the brain*". D12 summarized the interaction between the bot and user as ""*Expecting the unexpected*".

Tools used to develop bots: The developers had difficulties in finding tools for developing bots. D24, for instance, answered: "*No good frameworks existed for developing Slack bots in Python. I created one (...)*". The difficulty in using the tools were also mentioned in D1's and D2's answers. D1 reported that "*Integrating with third party APIs; orchestration*" is a hard issue, whereas D2 spotted that "*Integration with services, lack of api or documentation*" are a source of concern.

Testing Bots: We gathered evidence that testing is a major concern in bot development. Part of this is due to the intrinsic nature of bots: they have to be integrated with other applications to exist. D16 reported that "*They may be difficult to test on the API calls of the integrated software, since the test API calls may be limited to a simple ping*". Along these lines, D18 highlighted that "*Debug and connect each component*" can be a challenge. Among these applications, the most cited were: Slack, GitHub and Discord, alongside others like Salesforce, Microsoft and IBM Speech-to-Text/Text-to-Speech services and ERPs. In other words, bots are integrated with both open-source and propietary software.

Information Seeking: Another common concern identified is related to the lack of books, guidelines, or documentations related to bot development. To illustrate, D11 highlighted that "*[there is a] Lack of information... [we] need more open sources. e.g. [a] lexical analyzer"*. D15 complemented saying that "*Tutorials [are needed]*". This also means that our informants spend a lot of time seeking for information about bot development.

3.4. RQ4: What motivated developers to develop his(ers) most recent bot?

When analyzing the answers to the open questions about motivation for bot development, we identified four main categories: personal need, workplace assistance, personal learning and commercial purposes.

Personal needs: Some bots are created just to attend a personal need, as reported by D2: "*As i'm creating my own house automation box, i wanted a bot to interact with for a lot of purposes: weather, light, ...*". Although some bots were initially designed to deal with specific problems, they can be used for general public, as reported by D11: "*The weather in Korea is very volatile. I made it to know the weather. It notifies the user about weather every day specific time*".

Workplace assistance: In this situation, the bot was intended to be used on the workplace environment of the developer. D16 commented that (s)he wanted to facilitate the access to messages exchanged with the client (considering that the team uses a different communication channel from the one used in the client): "*We are using Intercom in my job for communication with clients, and Discord for in-team communication. There was a need for the Intercom messages to be easily visible to the team, without leaving Discord, in order to handle bug reports and requests better*".

The bots developed could also be used to automate tedious tasks, as D20 and D24 reported in their answers, respectively: "*Ops automation in team*", "*Automate tasks at my current workplace (which uses Slack) and test drive my own bot development framework for Python*".

Learning: In this case the developers simply wanted to gain experience and knowledge about bot development. As an example, D19 reported: "*for fun, and to learn something new*". Also, the different landscape of bot development motivated D3 to develop a bot so that (s)he could "*learn Slack API, computer vision, and to assist with a game*".

Commercial purposes: In this category are developers that intended to commercialize their bots. D30 was very objective in his answer. He said: "*Commercial purposes*". In this category, there is also the case of a developer being paid to develop a bot, as is the case of D34: "It was developed for a big company".

4. Discussion

Overall, our results suggest that bot development is a fairly recent activity among our respondents. 79% of the respondents have 5 or more years of software development experience while only 27,9% (11 developers) have more than 2 years of bot development experience. In particular, there is even a low amount of bots developed throughout this time: our respondents have developed 85 bots in total. One particular developer has developed 25 bots in the course of 9 years of experience, four have developed 10 to 12 bots and the other six have developed 2 to 5 bots.

When looking at the initial motivation for bot development (Figure 1-(a)) and for the most recent bot, we can notice that the main motivations are extrinsic, i.e., bot developers are motivated by external factors including automating tasks at work, addressing personal needs, etc. However, we also perceived that a smaller percentage of developers started implementing bots due to personal reasons (e.g., learning).

Another interesting result is noticed at RQ3, since five respondents claimed that they did not face any kind of challenges when developing their bots. When analyzing other answers from these particular respondents, we observed that four of them are *not* the most experienced software developers. However, one of these four (who has developed 2 bots, has 7 months of bot development experience, and develops bots as a hobby), stated that "*I personally feel that developing bots are not difficult since the programming interface are relatively easy. However, it may be intimidating for people starting out since it's different from conventional mobile or web app development.*". The other respondent that saw no difficulties in bot development argued that bot development was easy. In fact, he reported a staggering amount of 60 bots developed over the course of three months of bot development experience! In this case, we wonder what is the definition of a "bot" and "bot developing" for this particular respondent (e.g., is an automatic meme generator a bot?). An in-depth understanding about "what is a bot" was out of the scope of this study. However, given this respondent, we believe that this could be a potential follow-up study.

The results show that 39.5% of the difficulties faced by developers are related to the implementation of how the bot should interact with the user. This might be connected to the fact of the increasing demand of more intelligent bots with better decision making features [Wessel et al. 2018]. Previous work has shown that bots are used by developers for different ends of entertainment or to help on their work [Lin et al. 2016], our results provide similar results since most of our respondents created their bots with the intention of using them by themselves whether on a daily basis or at their workplace.

Finally, another challenge worth noticing is related to the bot testing, i.e., the process of testing the bots under development. When we analyzed how the bots were developed, we noticed that 50% of the bots designed to be used by groups were developed by only one person. Also, 87.5% of the bots that could be used by either groups of people or by an isolated user were developed by only one person. These results are reported at the end of section 3.2. This situation may have reflected on the testing difficulties stated by some participants (**RQ3**) since there is no better way to test a tool designed for collaborative work than with, indeed, a group of people. In fact, Grudin [Grudin 1994] has recognized this as an important issue in software for groups 25 years ago.

Still regarding to software testing, it is important to note that bot development may need a richer programming arsenal to deal with the "unexpected" behaviors, which is often the case when implementing chatbots, one specific type of bots. For instance, libraries that implement machine learning and artificial intelligence techniques are the foundations of bot development (although hardly discussed in, say, traditional software engineering courses). Even though some of these challenges have been anticipated by related work (e.g, [Arpteg et al. 2018]), we believe that much is still pending to be used in practice, in terms of for instance the improvements on software tooling to better aid developers or in better documentation (e.g., cookbooks, APIs, QA websites) that could better support software developers learning in this regard.

5. Limitations

First, although we obtained interesting results, we cannot generalize our characterization of the development community. We are limited by (1) developers that use GitHub, and also to open source projects selected when searching for the "slack bot" topic. It does not cover bot developers that host their projects on other websites or do not show their interest on bot-related projects on GitHub. This restriction may have altered the results as we chose a specific percentage of the bot development community to analyze. Second, another threat is related to the low amount of answers we received. Some of the reasons that might justify our low response rate include: (1) developers might have considered our questionnaire too long to be answered, (2) some users probably categorize 'BCC' emails as spam, (3) other users might not have any experience at all in bot development and decided not to answer it (since most of the emails we sent were to users that only stared a project), and even (4) due to the frequent use of questionnaire as a research tool nowadays. As pointed out in a blog post[4], developers are being swamped with survey invitations. In particular, some developers are indeed asking to remove their email addresses from open databases such as GHTorrent [Gousios 2013] in order to not be selected to participate in these studies anymore. As one potential mitigation plan, at the very least, in our survey we stated that the participation was voluntary and all personal information provided would remain anonymous. However, our final response rate was about 2%. As a lesson learned, we perceived that this approach of inviting a large number of software developers to participate in a survey might not be feasible anymore, and other approaches (e.g., using sampling algorithms to reduce the need to cover a wide range of software developers) should be explored. Finally, the analysis of the open questions was conducted individually. Due to the qualitative nature of this process, chances are that other topics could have been emerged if others have performed this analysis. To mitigate such bias, while one author was responsible for deriving the main themes, another author revised the work and, when needed, potential conflicts were discussed until agreement was reached.

[4] http://www.gousios.gr/blog/Issue-thirty-two.html

6. Related Work

Bots can be a valuable tool in the software development process, with them, repetitive and tedious tasks can be quickly resolved [Storey and Zagalsky 2016a]. Although bots have gained more prominence in recent years and provide advantages for developers to be able to market them [Klopfenstein et al. 2017], we find few cases of bot developers who are looking to market their bots or who are hired to develop a bot. Wessel and colleagues [Wessel et al. 2018] analyzed bots that automate tasks such as checking license functions, code quality, and testability. Similar to us, they focused on projects hosted on Github, but they studied how integrators (people who install the bots) and developers interact with projects that use bots. Something reported in their results was the demand for smarter bots. This may be related to the bot development difficulty discussed in RQ3, where 37 % of bot developers replied had bot implementation difficulties, mainly in the area of artificial intelligence and machine learning.

In our research we have seen several reports from developers who have started to develop bots to automate tasks. Furthermore, a considerable portion of them have already developed a bot to aid it in their work environment or in their daily activities. This is aligned with previous work from Lin and colleagues [Lin et al. 2016] who looked at developers who adopted Slack. One of their research questions was: Which bots do developers use and why do they use them? The results of this research reveal that software developers use bots to assist them at work or at their leisure.

7. Conclusions and future work

This research revealed that our participants are people that already have some experience in software development e have shown recent interest in bot development. The bots developed are mostly used by few users and had the main purpose of satisfying the developer's personal needs as well as to solve a problem in his workplace or to assist on his daily life, which means there is a low interest in commercial use. Nonetheless, the task of implementing a bot has revealed to be an activity that can present complicated tasks for the participants mostly when it is related to machine learning. Another problem also mentioned with certain frequency regarded the use of APIs of bot development whereas a huge percentage of the developers reported having difficulties using them.

For future work we plan to conduct semi-structured interviews with participants that answered our questionnaire and additional bot developers. Since integration was regarded as a major difficulty by our informants, we also plan to analyze the source code of the bot projects to understand how they integrate the bot projects within the non-bot projects.

Acknowledgments

We thank the 43 bot developers who participate in our survey. This research has been partially funded by the Brazilian National Council for Research and Development (CNPq), under research grants 420801/2016-2 and 311256/2018-0.

References

[Arpteg et al. 2018] Arpteg, A., Brinne, B., Crnkovic-Friis, L., and Bosch, J. (2018). Software engineering challenges of deep learning. In *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 50–59.

[Bradley et al. 2018] Bradley, N. C., Fritz, T., and Holmes, R. (2018). Context-aware conversational developer assistants. In *Proceedings of the 40th International Conference on Software Engineering*, ICSE '18, pages 993–1003, New York, NY, USA. ACM.

[Gousios 2013] Gousios, G. (2013). The ghtorent dataset and tool suite. In *Proceedings of the 10th Working Conference on Mining Software Repositories, MSR '13, San Francisco, CA, USA, May 18-19, 2013*, pages 233–236.

[Grudin 1994] Grudin, J. (1994). Groupware and social dynamics: Eight challenges for developers. *Commun. ACM*, 37(1):92–105.

[Klopfenstein et al. 2017] Klopfenstein, L. C., Delpriori, S., Malatini, S., and Bogliolo, A. (2017). The rise of bots: A survey of conversational interfaces, patterns, and paradigms. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, pages 555–565, New York, NY, USA. ACM.

[Lebeuf et al. 2018] Lebeuf, C., Storey, M., and Zagalsky, A. (2018). Software bots. *IEEE Software*, 35(1):18–23.

[Lebeuf et al. 2017] Lebeuf, C., Storey, M. D., and Zagalsky, A. (2017). How software developers mitigate collaboration friction with chatbots. *CoRR*, abs/1702.07011.

[Lin et al. 2016] Lin, B., Zagalsky, A., Storey, M., and Serebrenik, A. (2016). Why developers are slacking off: Understanding how software teams use slack. In *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion*, CSCW '16 Companion, pages 333–336, New York, NY, USA. ACM.

[Pérez-Soler et al. 2017] Pérez-Soler, S., Guerra, E., de Lara, J., and Jurado, F. (2017). The rise of the (modelling) bots: Towards assisted modelling via social networks. In *Proceedings of the 32Nd IEEE/ACM International Conference on Automated Software Engineering*, ASE 2017, pages 723–728, Piscataway, NJ, USA. IEEE Press.

[Storey and Zagalsky 2016a] Storey, M.-A. and Zagalsky, A. (2016a). Disrupting developer productivity one bot at a time. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2016, pages 928–931, New York, NY, USA. ACM.

[Storey and Zagalsky 2016b] Storey, M.-A. and Zagalsky, A. (2016b). Disrupting developer productivity one bot at a time. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2016, pages 928–931, New York, NY, USA. ACM.

[Strauss and Corbin 1990] Strauss, A. and Corbin, J. (1990). *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. Sage Publications, Newbury Park, California.

[Turing 1950] Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236):433–460.

[Wessel et al. 2018] Wessel, M. S., de Souza, B. M., Steinmacher, I., Wiese, I. S., Polato, I., Chaves, A. P., and Gerosa, M. A. (2018). The power of bots: Characterizing and understanding bots in OSS projects. *PACMHCI*, 2(CSCW):182:1–182:19.