Comparison of Open-Source Software Release and Civic Hackathon Organizational Structures regarding Sustainability Factors

A Research Paper submitted to the Department of Engineering and Society

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Andy Tan

Spring 2022

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

Kathryn A. Neeley, Associate Professor of STS, Department of Engineering and Society

Introduction

Civic technologies, information and communications technologies, have the potential to greatly benefit citizens' daily lives and improve citizens' confidence in public administration by improving the quality and clarity of public services, increasing citizen involvement in public decision making, and promoting information sharing (Boehner & DiSalvo, 2016). However, civic technologies continue to have a low perceived value and sustained use in public organizations (Hou & Lampe, 2017, p. 28). According to Skarzauskiene & Maciuliene (2020), most civic technology initiatives have failed to "yield innovative solutions, the consensus among stakeholders, or even collective action of any kind" (p. 3). Civic hackathons, time-limited software development competitions, have been hosted by both governments and private actors to bring together diverse groups of people to address societal challenges (Ermoshina, 2018; Gordon & Lopez 2019). However, many of these hackathons' resulting software projects are neither widely adopted nor sustainable. One study reported that about 65% of projects are discontinued after a hackathon, and only about 17% of projects continue to be worked on after a week past the end of a hackathon (Nolte et al., 2020, p. 10). With few coding projects surviving past the end of hackathons, the full potential of civic hackathons and their resulting civic projects have yet to be fully realized and don't justify the investments made in them (Medina Angarita & Nolte, 2020, p. 51). Participants also risk losing interest and motivation for participating in future civic projects. In order to gain insight into how hackathons can produce more sustainable software projects, this paper compares the general organizational structures of hackathons to the organizational structures of successful open-source software projects, a field with more examples of mature and sustainable software projects, in order to surface issues with the organizational structure of civic hackathons that hinder the development of sustainable software.

Problem Domain

Hackathons can be viewed as a software development environment that leverages the benefits of sports competition formats to attract a variety of people with different backgrounds. Hackathon organizers, which can consist of private sector or public sector organizations or individuals, are in charge of setting the rules of the competition, attracting sponsors, advertising to attract potential participants, and determining events and workshops, gathering and appointing mentors and judges in the process (Dainotti et al., 2018). Major League Hacking, a hackathon league that supports student hackathon organizers, also provides many online resources and guidelines for hackathon organizers. Using these resources, organizers can learn how to host similarly formatted hackathons using shared practices, such as holding hackathons over 24-48-hour periods during weekends (Major League Hacking, 2021). Although hackathons are relatively newer software development environments, most hackathons share a consistent base format.

As a software development environment, hackathons attempt to create sustainable software as one of their primary goals. Sustainability in the context of software development means the ability for a software project to be deployed and maintained for an extended period of time (Hou & Lampe, 2017; Knutas et al., 2019). Sustainability entails, among other requirements, obtaining some kind of financial support for a project. This use of software sustainability should not be confused with ecological sustainability. In order for a software project to be successfully sustained, it must first be adopted. Adoption means that the software project is able to attract enough users from its target user base to generate enough value to justify financially supporting the project. However, projects can be funded from a variety of sources, such as through public government funding or crowdfunding (Gonzalez-Cacheda, 2021). In

order for a hackathon project to be successfully adopted and sustained, it must be able to reach and be used by a broad audience of users, which is especially important for civic hackathons.

Civic hackathons are hackathons where the theme or focus is to build civic technologies, so they should especially prioritize sustainability. Civic hackathons attract people by advertising that participants are able to potentially affect their communities in lasting and meaningful ways, making it especially important for civic technologies to be sustainable. Civic technologies also usually have a large number of potential stakeholders, and adoption of a civic technology means that a significant number of citizens or organizational members interact with it, such that the project is valuable enough to keep maintained. However, civic hackathon projects are largely unsuccessful, as evidenced by a study reporting that 93.8% of the non-winners in a hackathon abandoning their projects (Gama, 2017).

It seems clear, then, that civic hackathons are unique development environments that have great potential for creating new impactful civic technologies, but their benefits are being underutilized. The resources used to develop software end up wasted, and hackathons "run the risk of being labeled as disingenuous engagement activities and also of exhausting participants' appetite for future involvement in new activities" (Johnson & Robinson, 2014). Civic technology also risks becoming undervalued by the public sector. In order to take advantage of civic technology's potential, instead of being relegated to simply providing "another outlet for the already engaged and vociferous" (May & Ross, 2018, p. 214), as well as wasting resources in developing unused civic tech, Yoshida & Thammetar recommend that the problem definition and development processes for creating civic tech should involve more product stakeholders in the problem definition process and place sustainability and accessibility as priorities when reconciling the different expectations of civic tech (2021, p. 56). Despite prior research into

technical sustainability factors of hackathon artifacts and development practices, the sustainability practices have yet to be applied in practice in hackathons. The hackathon outcomes indicate that there is a gap in research on how the hackathon's organizational structure and hackathon format influences and encourages software sustainability practices used by development teams during these events.

In order to gain insight into what roles might be missing or unbalanced in hackathon software development ecosystems, hackathons can be compared to open source software projects and the actors surrounding their release and maintenance, because the field of open source software is a more established paradigm that has also faced similar adoption and sustainability challenges. Similar to how hackathon projects are often abandoned after hackathons, there is evidence that a majority of open source software projects struggle to attract contributors (Hauge et al., 2010, p. 24). However, open source software is an older, relatively more mature field than hackathons, which has produced more examples of sustainable and widely used projects, including Mozilla browser, Linux operating system, Apache Web server, MySQL database, PostgreSQL database, PHP scripting language, Perl scripting language, Python scripting language, and GNOME desktop environment (Yang & Wang, 2008). The adoption of open source software is also a primary focus in existing open source software research (Hauge et al., 2010, p. 17). Additionally, open source software project development is not much different from other forms of software development. The field of open source software should be seen as an opportunity for investigating general software engineering challenges (Hauge et al., 2010, p. 35). Therefore, open source software is a solid foundation for analyzing sustainability challenges in a less mature software engineering environment, such as hackathons.

Prior research has identified multiple adoption factors for both civic hackathon projects and open source projects that are similar, as they are both software development environments that aim to reach large user bases. Civic technology platforms are mostly oriented toward citizens and governmental organizations (Skarzauskiene & Maciuliene, 2020). In order for both development environments to generate technologies that properly target these large audiences, prior research has been done to identify several adoption factors. Sánchez et al. (2020) have identified 22 different factors that have influence during the decision-making process for adopting open source software: 9 technical factors, 9 organizational factors, and 4 economic factors. Kilamo et al. (2012) also have proposed a 4-dimensional readiness evaluation with multiple measures for determining a project's readiness for open source release such that it will be likely to attract a community. The technical factors such as compatibility, reliability, usability, customization, documentation, maintainability, trialability, and so on align closely with the technical factors for civic hacking project adoption: increasing ease of understanding, use, and maintenance. For example, Hou and Lampe (2017) suggest reducing civic project complexity and increasing transparency by using low-tech solutions to increase external parties' understanding during the development process. Both areas of research also agree that software is more sustainable and readily adopted when organizational and economic factors such as the availability of support systems, the existence of partnerships with sponsors or political actors, and the strength of lasting relationships with stakeholders (Medina Angarita & Nolte, 2020; Sánchez et al., 2020). The adoption factors are similar between open source software releases and hackathons, so the roles required to achieve these factors to increase the likelihood of adoption are also similar. However, the responsibilities to appeal to these factors are assigned to different actors in each system.

When releasing software as open source, the publishing entity, or releasing authority attempts to attract individuals to form an open source software community, involving a variety of actors. Open source software communities consist of multiple layers of members, and individuals can be categorized depending on the type and frequency of individual contributions into these layers, which range from users to core development members (Kilamo et al., 2010; Young et al., 2021). Because users typically filter into and fill more significant contributor roles over time, this paper will group the intermediate layers with users and primarily reference the spectrum's two extremes of active core developers and users for simplicity. These community members are often drawn from and shared with existing open source communities of other projects. In addition to appealing to members of the existing open source communities, the releasing authority needs to also consider potential partners and interested parties, as well as choose between different licensing models and costs (Kilamo et al., 2012). Open source software releases and hackathon events share common responsibilities which are distributed among different actors, so the distribution of roles can be compared between them to reveal any missing or misallocated roles within hackathon events that lead to fewer sustainable outcomes.

Methods

As the previous section demonstrated, prior research has investigated key factors related to the adoption of both civic technology projects and open source software, and both sets of factors closely mirror each other. Both fields of research note increasing the ease of newcomers to understand, use, and maintain the projects, through documentation and reducing complexity. Both fields also note that developing strong, lasting relationships with stakeholders and financial sponsors often contributes to the sustainability of a software project (Sánchez et al., 2020; Hou, & Lampe, 2017). There has also been research with a focus on hackathon outcome sustainability,

but research has primarily focused on the sustainability of tangible hackathon artifacts and the processes performed by project teams, while also indicating a gap in research in investigating intangible hackathon outcomes, such as networking, learning, and connections with stakeholders (Medina Angarita & Nolte, 2020). Prior research also suggests that hackathon's time format may affect the sustainability of projects by influencing practices in the implementation and design phases of civic hackathon projects (Gama, 2017). However, there is a gap in prior research in addressing the impact of the organizational structure of hackathons and hackathon organizer's full potential impact on sustainability outcomes.

This paper differs from prior research by using the organizational structure of hackathons as a foundation for comparing the organizational structure surrounding open source software releases. It focuses instead on the distribution of roles and responsibilities in each system. By investigating the roles of other actors than only team members and the tangible artifacts they produce, insight can be gained into how the responsibilities can be better balanced between the groups of actors.

The primary research method was analyzing the differences between the sociotechnical system surrounding civic hackathons and the system surrounding active open-source software projects using multiple analogies. Analogies can provide insight when analyzing new technologies by comparing them from various focal points with previous technologies. However, each technological system is inherently complex, so multiple analogies are needed to adequately compare the two systems (Schwarz-Plaschg, 2018). Using actor-network theory to frame each system, the primary comparisons will be made between differing roles between actors and supporting organizational structures using the common factors for adoption as a starting point.

The analogy method is appropriate because both civic hackathons and open source communities can be viewed as organizational structures that aim to generate sustainable software. Civic hackathons are newer and do not have as much presence as open-source software in everyday use. For example, VS code is now the most popular source-code editor according to a survey performed in 2021 ("Stack Overflow Developer Survey 2021," 2021). Although civic hackathons are varied, most modern hackathons follow guidelines commonly shared among hackathon organizers, and have the same groups of actors. Likewise, open source software projects also have a lot of variance between communities, but have actors that can be commonly classified among them. The common features between individual projects, their adoption factors, and their surrounding communities and organizations enables open source projects and civic hackathon projects to be compared using analogies.

The discussion will primarily be drawing information from literature reviews for civic hackathons, that have analyzed multiple projects within each hackathon in order to understand the common organizational structures across multiple hackathons. Literature reviews for open source software adoption will also be used, as those papers aggregate data and information from successful open source projects and have formed models for presenting common structures across successful open source communities.

Results/Discussion

Both hackathons and open source software communities can be seen as organizational structures that promote creating software projects. An open source software project's releasing authority has most of the responsibilities in developing the software before and shortly after it is released. The primary phases of the software development lifecycle, common among various software development methodologies, are planning, design, development, testing, deployment,

and maintenance. These phases are assigned to the releasing authority before a project is released. After it is released, an open source software project's planning, design, development, and maintenance responsibilities are passed on to the core developers, and testing and deployment responsibilities are passed to the users. However, the releasing authority typically maintains close ties with the core developers and can also influence the planning, design, and development direction of the project (Kilamo et al., 2012). While the releasing authority has many responsibilities, it is able to perform them, financial concerns aside, because there is no time constraint, unlike with the hackathon teams.

When releasing open source software, the releasing authority's role of eliciting requirements from users is analogous to the role of team members developing projects in civic hackathons. When comparing the two actors, both need to plan and design a product. However, the team members are often not required by the event rules and, by extension, the organizers to meet with stakeholders explicitly before or during the competition, so the responsibility falls on the team members to reach out to stakeholders (Major League Hacking, 2021). However, due to the time limited format, software design processes must be performed within a "condensed time frame," making it easy for teams to overlook proper requirements elicitation practices (Flux & Hurst, 2021, p. 15). According to Gama, "there is evidence to support the claims that civic apps are based on developer experience, instead of being based on actual citizen needs" (2017, p. 17). Hackathon organizers ultimately determine the format of each hackathon, including the time limit, so they have the power to influence the stages of the software development process.

The responsibility of determining the requirements and interacting with stakeholders and users falls with the development team with little direct intervention by hackathon organizers outside of connecting team members with mentors during the ideation phase. Ideas that are

generated before a hackathon begins can be used for projects, and issues that projects address do not have to be original. The main restriction is on the time allotted for creating the design and implementation of a project, which must be started during the hackathon according to Major League Hacking hackathon organizer guidelines (2021). With open source software, the responsibility of interacting with users to determine requirements lies with the releasing authority and the development team it appoints. However, without a time limit, the releasing authority is able to adequately assess the user's needs and pain points before and during the implementation phase.

In order for a hackathon project to adequately assess requirements for the specification elicitation phase, the burden of responsibility to interact with stakeholders should be shared with other actors such as the hackathon organizers. The hackathon organizers are involved with hackathon tasks before the day of the hackathon, and are best suited to either gather or delegate the task to other actors, such as mentors, before the hackathon starts. Mentors would be suited to this task because they typically advise teams during the ideation phase of hackathons. Regardless, interacting with stakeholders beforehand, or having the hackathon organizers attract more stakeholders to interact with the groups during the event appears to be an option to rebalance responsibilities such that the actors with the most time constraints do not have the most responsibilities. Extending the time dedicated to determining specifications to allow for more precise specifications would also be a potential way to address the issue. The current recommended rules provided by Major League Hacking (2021) do not require that ideas be created during the ideation phase. Instead, they only require that the project design and implementation be completed during the event. The hackathon organizers have the power to change the format of the hackathon to facilitate more time spent on determining requirements, by

making separate judging and reward pools for better prepared ideas, or by organizing communication and team formation between participants before the typical implementation window.

In civic hackathon projects, the project team is analogous to both the development team of a releasing authority and the core developers of an open source project because both are the primary technical experts that have the responsibility of designing and developing their projects. Additionally, releasing authorities and hackathon project teams are both responsible for preparing support infrastructure. Hackathon participants are capable of producing sustainable software (Medina Angarita & Nolte, 2020). Therefore, there is little difference in the sustainability of outcomes due to differences in abilities of open source software developers and hackathon project developers. Instead, the differences stem from environmental factors, such as the time limit of the competition or the process of team formation, are likely causes for why many projects are incomplete or abandoned by the end of the hackathon (Nolte et al., 2020). Civic hackathon organizers are also able to limit team size in hackathon rules, which would promote smaller teams with greater development efficiency.

The role of hackathon organizers is primarily to organize the competition; they foster forming teams between individuals and attract sponsors, judges, and mentors. This is similar to how a releasing authority organizes its own development teams, as it also needs to attract sponsors and find metrics for evaluating the success of an open source release. The hackathon organizers also organize the judging and format of the competition, meaning they can be viewed as a temporary free advertisement provider for teams to showcase a demonstration of their projects to interested parties. The organizers fulfill the roles of attracting interested parties and demonstrating projects to external parties to a limited extent, but often do not continue to do so

for projects after the end of the hackathon. However, with open source software, the releasing authority continuously monitors the software, because it remains a stakeholder in the success of the project.

While stakeholders in civic hacking projects are public governing bodies and citizens or non-governmental organization members, the users in open source communities can hold any of these roles and more. Popular open-source projects are often platform technologies that appeal to a wide user base and serve as a dependency for many other projects. Ultimately, open-source projects aim to leverage the benefits of having users contribute to developing and maintaining them, or to at least have more exposure for their projects (Young et al., 2020). Similarly, the development and maintenance of hackathon projects would also likely be more successful with more support from more users. With more users, the potential for crowdsourcing maintenance work or direct crowdfunding becomes more feasible for benefitting the sustainability of a project.

The open source software releasing authority provides core members, or experts, whereas the hackathon projects are not expected or required to maintain the projects after they achieve a minimum viable product. Adequate documentation such that an outsider could understand how to use and extend the code is encouraged, but not required by hackathons (Major League Hacking, 2021). There is no clear actor that is assigned the role of maintaining the project after the end of the hackathon. Whereas with open source software, the releasing authority has to provide support infrastructure and core members until the project becomes self-sustaining. Until then, and possibly even after, the releasing authority cannot hand off its role as the primary maintainer of the project. Therefore, as the technical experts of their projects, hackathon project

members should also be encouraged to maintain their projects after the end of hackathons, until they are able to hand the maintenance off to a partner or sponsor.

Unless the organizers are potential sponsors, in the case of government held civic hackathons, there is no guarantee of funding or sponsorship for most projects. Many non-winning teams choose to abandon their hackathon projects (Gama, 2017). The sponsors for the hackathon influence the competition's main theme and goals of the projects, but hackathon organizers are not expected to find sponsors for every individual project. It is also unlikely that sponsors would be willing or able to shoulder the responsibility to do so either, considering that a single hackathon can have hundreds of submissions. Hackathon organizers also cannot accept too many sponsors or they will risk devaluing the position of sponsor (Major League Hacking, 2021). Therefore, the responsibility is on the team members to find and attract other sources of funding. The difficulty in advertising individual projects does not appear to be able to be remedied with shifting responsibilities.

Conclusion

Comparing the organizational structures between hackathons and open source software releases reveals that nearly all of the responsibilities assigned to a releasing authority have been left to hackathon project teams to perform in a condensed time frame, while the responsibilities could be more evenly shared between both organizers and participants, as both have qualities that mirror releasing authorities. Hackathon organizers can take on or delegate more responsibilities usually held by the releasing authority of an open source project, notably the responsibility to adequately interact with and elicit requirements from stakeholders. Hackathon organizers can also change how teams are formed to encourage that the teams are willing to

maintain their own projects after the end of a hackathon. Hackathon organizers have the potential for having a greater impact on sustainability outcomes than they currently do.

This research was limited because it only used successful open source software projects as a baseline for comparison. Future work should attempt to examine examples of both successful and unsuccessful open source project releases, as well as both successful and unsuccessful civic hackathon outcomes, although this may be challenging, as unsuccessful cases are often not as well documented. Additionally, future research should directly investigate the effect of varying competition rule parameters on the sustainability of hackathon outcomes, which involve directly interacting with hackathon organizers in multiple hackathons. This research was also limited because it did not consider differences in hackathon themes, which may be more compatible with prioritizing sustainability. In order to address these differences, future research should investigate the effect of various hackathon themes with sustainability outcomes.

References

Boehner, K., & DiSalvo, C. (2016). Data, Design and Civics: An Exploratory Study of Civic Tech. *34th Annual Chi Conference on Human Factors in Computing Systems, Chi 2016*, 2970–2981. https://doi.org/10.1145/2858036.2858326

Dainotti, A., Katz-Bassett, E., & Dimitropolous, X. (2018). The BGP hackathon 2016 report. *ACM SIGCOMM Computer Communication Review*, *46*(3), 12:1-12:6. https://doi.org/10.1145/3243157.3243169

Ermoshina, K. (2018). Civic Hacking Redefining Hackers and Civic Participation. *Tecnoscienza-Italian Journal of Science & Technology Studies*, *9*(1), 79–101.

Flus, M., & Hurst, A. (2021). Design at hackathons: New opportunities for design research. *Design Science*, *7*. https://doi.org/10.1017/dsj.2021.1

Gama, K. (2017). Preliminary Findings on Software Engineering Practices in Civic Hackathons. *2017 IEEE/ACM 4th International Workshop on CrowdSourcing in Software Engineering (CSI-SE)*, 14–20. https://doi.org/10.1109/CSI-SE.2017.5

Gonzalez-Cacheda, B. (2021). An approach to the participative limitations of civic crowdfunding. *Idp-Internet Law and Politics*, *33*. https://www.webofscience.com/wos/woscc/full-record/WOS:000692662900001

Gordon, E., & Lopez, R. A. (2019). The Practice of Civic Tech: Tensions in the Adoption and Use of New Technologies in Community Based Organizations. *Media and Communication*, 7(3), 57–68. https://doi.org/10.17645/mac.v7i3.2180

Hauge, Ø., Ayala, C., & Conradi, R. (2010). *Adoption of open source software in software-intensive organizations – A systematic literature review*. Information and Software Technology, 52(11), 1133–1154. https://doi.org/10.1016/j.infsof.2010.05.008

Hou, Y., & Lampe, C. (2017). Sustainable Hacking: Characteristics of the Design and Adoption of Civic Hacking Projects. *Proceedings of the 8th International Conference on Communities and Technologies*, 125–134. https://doi.org/10.1145/3083671.3083706

Johnson, P., & Robinson, P. (2014). Civic Hackathons: Innovation, Procurement, or Civic Engagement? *Review of Policy Research*, *31*(4), 349–357. https://doi.org/10.1111/ropr.12074

Kilamo, T., Aaltonen, T., & Heinimäki, T. J. (2010). BULB: Onion-Based Measuring of OSS Communities. In P. Ågerfalk, C. Boldyreff, J. M. González-Barahona, G. R. Madey, & J. Noll (Eds.), *Open Source Software: New Horizons* (Vol. 319, pp. 342–347). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-13244-5_29

Kilamo, T., Hammouda, I., Mikkonen, T., & Aaltonen, T. (2012). From proprietary to open source—Growing an open source ecosystem. *Journal of Systems and Software, 85*(7), 1467–1478. https://doi.org/10.1016/j.jss.2011.06.071

May, A., & Ross, T. (2018). The design of civic technology: Factors that influence public participation and impact. *Ergonomics*, *61*(2), 214–225. https://doi.org/10.1080/00140139.2017.1349939

Medina Angarita, M. A., & Nolte, A. (2020). What Do We Know About Hackathon Outcomes and How to Support Them? – A Systematic Literature Review. In A. Nolte, C. Alvarez, R. Hishiyama, I.-A. Chounta, M. J. Rodríguez-Triana, & T. Inoue (Eds.), *Collaboration Technologies and Social Computing* (pp. 50–64). Springer International Publishing. https://doi.org/10.1007/978-3-030-58157-2_4

Nolte, A., Chounta, I.-A., & Herbsleb, J. D. (2020). What Happens to All These Hackathon Projects? Identifying Factors to Promote Hackathon Project Continuation. *Proceedings of the ACM on Human-Computer Interaction*, *4*(CSCW2), 145:1-145:26. https://doi.org/10.1145/3415216

Sánchez, V. R., Ayuso, P. N., Galindo, J. A., & Benavides, D. (2020). Open Source Adoption Factors—A Systematic Literature Review. *IEEE Access, 8*, 94594–94609. https://doi.org/10.1109/ACCESS.2020.2993248

Schwarz-Plaschg, C. (2018). The Power of Analogies for Imagining and Governing Emerging Technologies. *NanoEthics*, *12*(2), 139–153. https://doi.org/10.1007/s11569-018-0315-z

Skarzauskiene, A., & Maciuliene, M. (2020). Mapping International Civic Technologies Platforms. *Informatics-Basel*, *7*(4), 46. https://doi.org/10.3390/informatics7040046

Stack Overflow Developer Survey 2021. (2021). Stack Overflow. *https://insights.stackoverflow.com/survey/2021/?utm_source=social-share&utm_medium=social&utm_campaign=dev-survey-2021*

Yang, J., & Wang, J. (2008). Review on free and open source software. *2008 IEEE International Conference on Service Operations and Logistics, and Informatics, 1*, 1044–1049. https://doi.org/10.1109/SOLI.2008.4686552

Yoshida, M., & Thammetar, T. (2021). Education Between GovTech and Civic Tech. *International Journal of Emerging Technologies in Learning*, *16*(4), 52–68. https://doi.org/10.3991/ijet.v16i04.18769

Young, J.-G., Casari, A., McLaughlin, K., Trujillo, M. Z., Hébert-Dufresne, L., & Bagrow, J. P. (2021). Which contributions count? Analysis of attribution in open source. *2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR)*, 242–253. https://doi.org/10.1109/MSR52588.2021.00036