Time limited development

An overview of hackathons, the adaption to design and overview of
the most critical factors for time limited development

Anders Anvik

Department of Engineering Design and Materials

Norwegian University of Science and Technology

ABSTRACT

We are moving into a society where knowledge spreads in a much faster rate than before. Companies have less time to react to changes in the market and monopoly is not held for long. A way to compensate for this increasing demand is to find methods to reduce the time used, while still keeping a high enough quality. This review will take a look at events that can last up to a week, with special attention to 'hackathons´, which has become a familiar arena for software developers to show off their skills and do development in a very short amount of time, and how this phenomenon has been adopted to other areas of development, like design. This review will also look at the most critical factors involved in hackathons and time limited development to see what has already been covered by literature.

KEYWORDS: Time limited development, hackathon, designathon, design jam

1.   INTRODUCTION

The world around us is changing. We are moving into a society where design and product development is becoming much more rapid than before. Once it was possible for individuals and individual businesses to have a monopoly of knowledge over a longer period. However, this is no longer case. Today, knowledge spreads across borders with great speed [1]. Companies face challenges regarding creating new products to maintain a market advantage, to respond to competitors' swift actions and the fact that the life cycle of products is becoming increasingly shorter. Competitors and customers have become unpredictable and to get a competitive advantage in a fast moving society it has become important to have a rapid implementation from idea to market [2].

With classical development strategies, and even new methods like agile and lean development, it is hard to find room for creativity and wild exploration. The risk factors of starting a project with little research and experience behind is high. When combining this with the growing need to use less time from idea to market, companies look to use new methods to do fast explorations and rapid development. One popular method for

doing this within software development is called 'Hackathons´, a time limited event where programmers look to do 'quick and dirty´ development.

This review will take a look at hackathons, its uprising and what literature believe to be the most important factors for having a successful hackathon and end product. This review will also take a look at how hackathons is being adapted to other areas. This is to find out what factors and benefits from hackathons and other adaptations have been documented, when hackathons can be a useful method for development and to investigate whether this phenomenon already has been researched and documented in literature. The approach for finding this documentation was to search using known search engines for scientific papers, like scopus.com, but I have also used less academic searches through magazines etc. since the topic and phenomenon I am writing about is fairly new.

2. HACKATHONS

2.1 Origin

The term 'hackathon' is a combination of the concepts of 'hacking' and a 'marathon' It is a highly engaging and continuous gathering of programmers that collaboratively code in an extreme manner over a short period of time, usually 24-48 hours. The idea is for each developer to have the ability and freedom to work on whatever he or she wants [3], and strives to embody the tone of 'No Talk, All Action' [4]. The structure and characteristics of hackathons are usually similar, but can vary in purpose and execution.

It all started in the 1960s at MIT, where self-imposed hackathons where done by students who preferred working in 24-hour 'marathon bursts'. It wasn't until the late 1990s that the version of hackathons we have today emerged, where the focus for the end product is to create an application, or software program, that focuses on a specific programing language such as or an application programming interface (API). Hackathons can also focus on developing applications for particular platform such as mobile apps, operating systems, web systems or video game development. Since the mid-2000s, hackathons has become significantly more widespread and an arena for talented programmers to show off their skills in a short amount of time. It also became a scene for companies and venture capitalists to host competitions to create new technology and fund promising projects [5]. There are usually prizes involved to the winner of the hackathon, where the prize money can vary from a few hundreds and up to a million dollars.

Hackathons are also being done to adjust the point-of-view of employees in a company. By having a project run for a limited amount of time, you limit the risk taken. Companies then let employees take a new look and approach to the problem, all without any risk, and with little expectations of a viable product. By removing these risk factors around a project, you remove uncertainty and the fear of failing, which result in more innovative and creative results. Hackathon can be done as an external or internal event. Companies such as Microsoft, Google, Yahoo! and NASA organize hackathons externally [6], and have had success with this. RHoK (Random Hacks of Kindness) is a result of these companies initiative [7]. Facebook has become known for its hackathon culture and continuously organizing internal hackathon [2].

Hackathons has become a popular setting worldwide. There are different types of hackathons that focus on different aspects or topics. There are hackathons for; women-only, teens, college students, to fight autism, to improve education, to help veterans, to build Occupy Wall Street protest tools, on clean energy, on grocery shopping in Vermont, and 14 hackathons to troubleshoot water pollution [4]. Common for all of them is that it is a setting for programmers to do 'quick and dirty' development. It is not about getting top quality, but to get create a software program in a short amount of time that gives you an insight into the developers ideas. The hackathon setting is therefore used as a way to compensate for the increasing demand to reduce the time in development, as well as doing rapid prototyping.

2.2 Key characteristics

There are many different elements of a hackathon and 14 ingredients have been identified and considered important by Artiles (2013). These ingredients are presented in 'Table 1'. These ingredients can be seen as elements that are tweaked by organizers and sponsors of the event to achieve their goal and to best represent their theme for the event. When organizing an event like a hackathon, it is important to find the right question the event is trying to answer, and then the right setup can be chosen by mixing the 14 ingredients [4].

A conference paper named 'Hackathon – A Method for Digital Innovative Success: A Comparative Descriptive Study' has been peer reviewed in 2014, but the full paper has not yet been published. The aim of the paper is to identify factors leading to the success of hackathon contests. This has been done by

examining six such contests held between the years 2012 and 2014 and a total of six factors, reflecting the expectations held by both the hackathon organizers and its participants were identified. The results obtained showed that the identified factors are correlated to the success of such contests. However, the level of influence of each factor on the success of the contests differed in each case. As such, while each factor is of importance, they are all dependent on each another [8].

Element	Description
Overnight stay	The opportunities that participant has to sleep
Number of participants	Affects the support each participant gets as well as how many teams and projects are formed
Application process	Affects the barrier for participating
Award incentives	Ref. figure 2.
Mentorship provided	Support and feedback during event and keeping teams within the goal/theme of the event
Workshop offered	Efficiency of production of prototypes and testing
Theme delineation	Affecting the expected result
Challenge presentation	Helps participants to decide on what project they want to work on
Team formation	On what basis teams are formed
Project selection	On what basis projects are selected
Medium of deliverable	To force flow of work
End-user involvement	Can lead to reiteration and product endorsement
Amenities at the event	Things that contribute to physical or material comfort, ref. Figure 2.

Table 1: Important ingredients of a hackathon

2.2 Format

More often than not, the theme of the hackathon is known for the participants beforehand. This gives participants time to come up with relevant ideas related to the topic to work on. It also makes it easier for people to find the right people to cooperate with and create a team. This is done

in the pre-phase of a hackathon. Other important factors in the pre-phase are to be both physically and mentally ready for an intense work session. Getting enough sleep, having eaten enough food beforehand and having the right attitude going into the hackathon important and can be crucial for the end result. Hackathons can be characterized as an event where there is little to non-sleeping and being prepared is therefore very important.

the hackathon, there are representatives available for the hackers to consult with and get technical support. Typical consumption includes coffee, snacks and energy drinks. If the hackathon lasts only for a day, needs like food and sleep are often deprioritized to have as much time as possible to develop. At the end of the hackathon, teams present their idea/prototype in a demonstration and get an evaluation for their efforts.
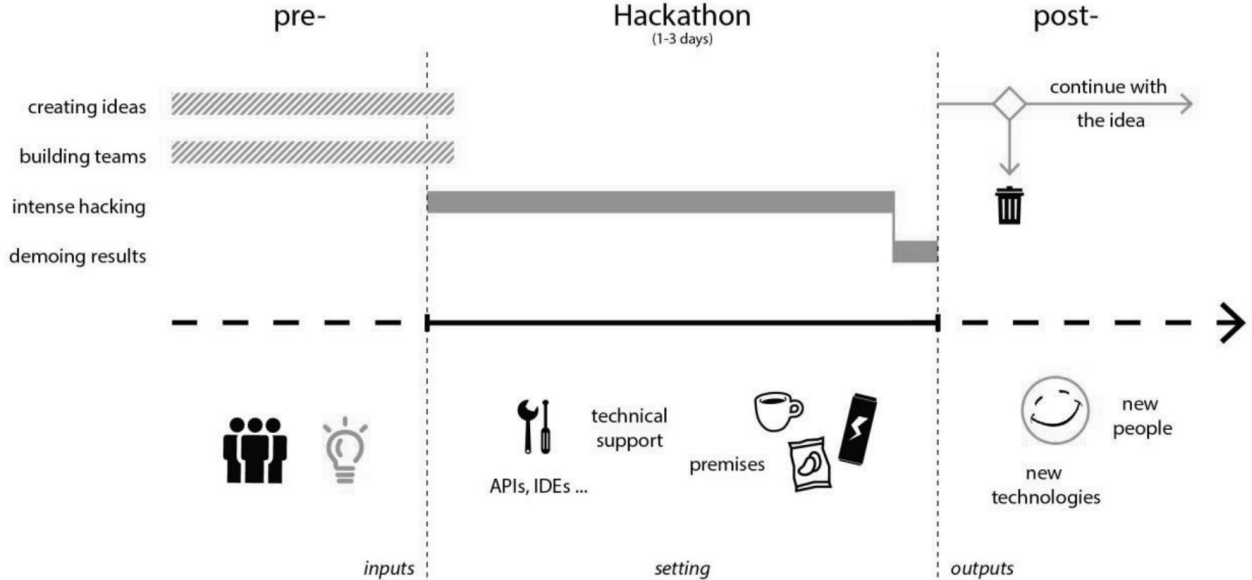

Figure 1: Typical timetable for a hackathon with its different phases [2].

The hackathon itself often starts with introduction to the event and the reason for hosting it. After that, people are free to form final teams and do idea generation together. Teams can also be organized by the organizers of the hackathon, based on the skills and interest of the participants. The time spent of creating ideas is often very short if the theme of the hackathon is known beforehand, and participants want to start programing right away. The challenge now is to connect the available data provided by the host to the idea, to create a prototype. During

In the post-phase of a hackathon, what happens to an idea or prototype depends on how well it does in the competition and the feedback it gets and if it is picked up by organizers and/or investors it can be taken to the next level of development. But what might be even more important than the result from the hackathon, is for the participants to get new connections, grow its own network and learn new skills. In 2012, TokBox asked 150 hackathon participants from across the United States of America their opinion on different aspects of hackathons they attended. When participants were asked why they attended, the two top reasons where learning (86%) and networking (82%). In 'Figure

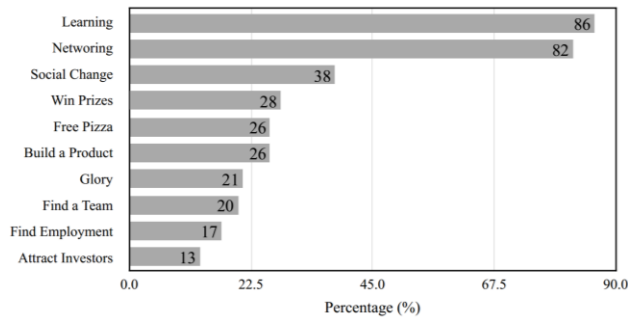2', the results for the reason to attending hackathons are summarized [9].

Figure 2: Graph of reasons for attendance [9].

3. ADAPTATION

As hackathons has spread wide, so has the participation from other disciplines. The word 'hackathon' has been adapted and broadened and it is believed that hackathons can very well be adapted by other areas than programing. 'For instance, 'hackathons' have become an important mechanism for identifying talent in software engineering, and similar events may provide that function for managerial talent.'[10] This statement can very well be said for many other disciplines. More and more non-software development companies have started to hold events where the objective is to either show off skills, do rapid concept development or both.

3.1 DESIGNATHON

A 'designathon' is a form of hackathon and is based on this phenomenon. The setup is very similar with the pre-, during- and post-phases and the 14 ingredients mentioned above. The only literature I found about designathons is based on a paper by Artiles and Wallace [4]. In this paper they describe an event held at MIT called the Education Designathon, where the object was to 'hack' the educational system. They

changed the name to 'desigrnathon' to attract a new audience to the scene of hackathons. This is to get the variety of participants up and different skills and knowledge contribute into a more whole end-product. What makes the designathon differ from other hackathons is that the content is broadened and the number of topics is expanded.

As humans we often fall within one of two spaces when problem solving: quick and weak solutions, or lengthy and worthy solutions. In the paper by Artiles and Wallace, it is suggested how an ideal designathon event would use the ingredients from a hackathon to move from a quick and weak solution toward quick and worthy results. This proposal is shown I 'Figure 3'. How to create an event that uses short amount of time and has highly promising quality is yet not described. It is clear that the quality of the solution increases the more time you spend, but the relationship between the 14 ingredients of a successful hackathon and time is not clear.
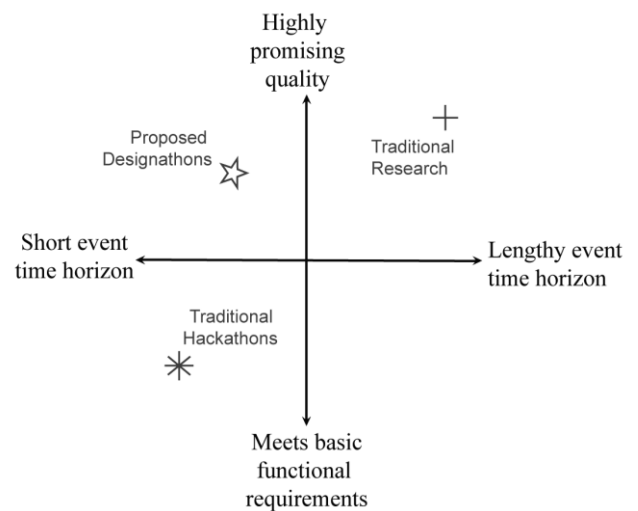

Figure 3: Regimes of results under different problem-solving events [4].

A relationship between how much time is spent on the problem and the quality of the solution

has been suggested as a logarithmic relationship, where the quality is a function of the log of time. This is represented in 'Figure 4'. The y-axis shows the quality of solution, where basic quality is described as a solution that meets functional requirements, but shows no depth in thought or idea generation. Advanced quality is a solution that shows depth of thought in problem identification, well thought out and researched ideas, and potentially some end-user generated feedback. Proven quality has a re-iterated and tested solution with results and evaluation before being released as a product [4].
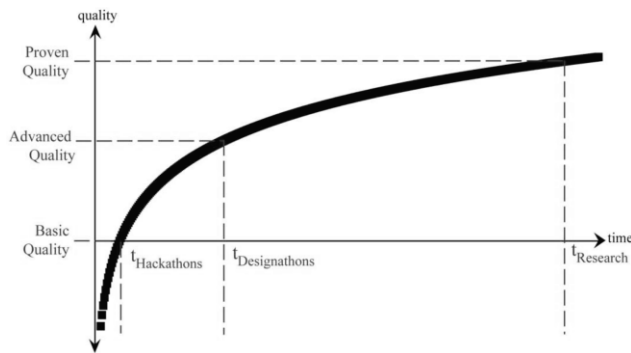

Figure 4: Relationship between quality and time under different problem-solving events [4].

3.2 DESIGN JAM

The event and term 'Design Jam' is an adaptation from the musical event 'jam'. A jam or jamming session is a musical event, process, or activity where musicians play (i.e. 'jam') by improvising without extensive preparation or predefined arrangements. Jam sessions are often used by musicians to develop new material (music), find suitable arrangements, or simply as a social gathering and communal practice session. Jamming sessions range from loose gatherings of armatures to events with coordinators the act as a 'gatekeeper' [9].

Design Jams work in just the same way as a music jam, just that it is not your instruments that you bring, it is your ideas. You bounce ideas off one another and build on what bounces back, and you turn your ideas into a concrete design and prototype, but there is one difference from hackathons and designathons. While hackathons and designathons has an early stage development focus, has jams the focus on the end-user [11]. This makes it so the design focus is more evenly distributed and you get a more complete vision of a product.

One example of a design jam is the 'Legal Design Jam'. A Legal Design Jam brings together a group of motivated individuals from different fields (e.g. designers, lawyers, policy-makers, coders, innovators, business people…) and, together, gives an extreme user-centric makeover to a legal document. The idea is borrowed from hackathons and service jams, and seeks to engage people to rethink and innovate the very concept of what a legal document should be, look and feel. During the event participants ideate and prototype a new version of a legal document, visualize and create layouts, rethink the structure and simplify its language [12].

In 2011, the Hasso Plattner Institute of Design at Stanford University (d.school) held a Design Jam with 200 participants. The objective of the Design Jam was to give an introduction to the five modes of designing (empathize, define, ideate, prototype and test [12]), all in 53 minutes. The topic of the jam was to 'Redesign the gift-giving experience, based on your partner's experience in giving gifts', and participants were given 14 minutes to empathize, 6 minutes to define, 18 minutes to ideate, 7 minutes to prototype and 8 minutes to test. The result from such an event might surprise. You can conduct an enormously beneficial, time-constrained design jam, which if done regularly is certain to shake loose some radical ideas [14]. This type of design jam and

time limited development is an example of a concept taken to the extreme.

4. CONCLUSIONS

Hackathons have become a known phenomenon and is well tried and the typical setup is well known. While I have found literature that identifies critical factors for a hackathons to be considered successful, the number of papers on this is few and I have found little to non-literature describing what the critical factors for adaptations of this phenomenon are. There has been studies where user questions has been answered prior to an event and analyzed to improve upon hackathons and designathons, but I have yet to find conclusive data on which critical factors are most important for a hackathon to be successful, and even less evidence regarding adaptations like designathons. I think the suggested relationships between quality and time shows potential, but it need more research.

Hackathons and its adaptations have shown that there is great potential to let people explore ideas and make products in a short amount of time without having to deal with the everyday performance stress and boundaries. Whether this phenomenon is suitable for solving companies increasing demand for productivity and getting a product rapidly implemented from idea to market is yet to be seen. If hackathons and other adaptations are to be efficient and prove to be productive for industries, more research needs to be put down to find what are the critical factors regarding such time limited and pressured development events.

Factors that I see as important and needs to be considered is the factor of time and how this affects the end product. The fact is that that you do not have a lot of time at each usual development stage, so which development stage is the most important and how is this weighted based on how much time one has at disposal. Other factors that I would like to know more about in this context is how does group size affect the process and end result, as well as how much preparation is done by the participant before the event and the effect of this.

REFERENCES

[1] Kjell A. Nordström, Jonas Ridderstråle, *Funky Business*, Oslo: Universitetsforlaget, 2006.

[2] Komssi, M., Pichlis, D., Raatikainen, M., Kindström, K., Järvinen, J. (2014). *What are hackathons for?* Aalto University, Finland

[3] Jansenn, C. *Technopedia*. Retrieved January 7, 2015, from Dictionary: http://www.technopedia.com/definition/23193/hackathon

[4] Artiles, J., Wallace, D. (2013). *BORROWING FROM HACKATHONS: OVERNIGHT DESIGNATHONS AS A TEMPLATE FOR CREATIVE IDEA HUBS IN THE SPACE OF HANDS-ON LEARNING, DIGITAL LEARNING, AND SYSTEMS RE-THINKING.* Massachusetts Institute of Technology, Cambridge, MA

[5] Leckart, S. (2012) *The Hackathon Is On: Pitching and Programming the Next Killer App*. Retrieved January 13, 2015, from Magazine: http://www.wired.com/2012/02/ff_hackathons/all/1

[6] Montalbano, E. *NASA, Microsoft, Google Hosting Hackathon*. Retrieved January 13, 2015, from: http://www.informationweek.com/applications/nasa-microsoft-google-hosting-hackathon/d/d-id/1089739?

[7] Random Hacks of Kindness. Retrieved January 14, 2015, from: http://www.rhokaustralia.org/

[8] Soltani P.M., Pessi K., Ahlin K. and Wernered I. (2014) *Hackathon – A Method for Digital Innovative Success: A Comparative Descriptive Study*. Retrived February 3, 2015, from

Gothenburg University Publications: http://gup.ub.gu.se/publication/206665-hackathon-a-method-for-digital-innovative-success-a-comparative-descriptive-study

[9] Briscoe, G., Mulligan, C. (2014). *Digital Innovation: The Hackathon Phenomenon*. Queen Mary University London, Imperial Collage London, United Kingdom

[10] Terwiesch, C., Ulrich, K. (2014). *Will video kill the classroom star? The threat and opportunity of MOOC for full-time MBA programs*. The Warthon School University of Pennsylvania, USA

[11] Musil, J., Schweda, A., Winkler, D., Biffl, S. (2010) *Synthesized Essence: What Game Jams Teach About Prototyping of New Software Products*.
Christian Doppler Laboratory for Software Engineering Integration for Flexible Automation Systems

[12] Legal Design Jam. Retrieved January 20, 2015, from: http://legaldesignjam.com/about/

[13] d.school. *The Bootcamp Bootleg*. Hasso Plattner Institute of Design at Stanford

[14] May, M. E. (2011) *How To Hold A Design Jam In 53 Minutes*. Retrieved January 20, 2015 from: https://www.americanexpress.com/us/small-business/openforum/articles/how-to-hold-a-design-jam-in-53-minutes/