The End of Mobile Software Engineering (As We Know It)

Robin Nunkesser
Hamm-Lippstadt University of Applied Sciences, Marker Allee 76–78, 59063 Hamm, Germany

Keywords: Software Engineering, Mobile Software Engineering, Research Agenda, Mobile, iOS, Android.

Abstract: Mobile Software Engineering as a separate research field has been relevant to speed up Software Engineering research in the area of mobile devices and applications. However, there are strong signs that the need for separate research in Mobile Software Engineering is coming to an end. This is not due to a decline in the importance of mobile devices and applications, but due to the impact of mobile on Software Engineering in its entirety. Existing mobile specific research may be reconsidered for its implications on Software Engineering in general. Future research can profit from considering mobile and non mobile devices and applications whereever possible.

1 INTRODUCTION

When writing about Mobile Software Engineering and its end, it feels necessary to explain that there was and still is a field of research called Mobile Software Engineering. The term *Mobile Software Engineering* is not as common as Software Engineering and it is not as well defined. However, it is a term that is used in the literature and in the context of conferences and journals.

The origin of Mobile Software Engineering is closely related to the success of mobile applications. An early milestone for application development for mobile devices was reached in 2001, when the Siemens SL45i allowed Java 2 Micro Edition (J2ME) applications to be downloaded with the Wireless Application Protocol (WAP). Until 2008, mobile application usage grew slowly but steadily. The opening of the iPhone App Store in July, 2008 marked a starting point for exponentional growth (Wasserman, 2010) with more than a billion application downloads in less than a year.[1]

The success of mobile applications naturally led to more considerations about the impact on Software Engineering (for early works, see most notably Wasserman (2010); Dehlinger and Dixon (2011)). Wasserman (2010) proposed a research agenda for Mobile Software Engineering and in the following years special journal issues such as Mobile Software Engineering and Systems and conferences such as the International Conference on Mobile Software Engineering and Systems (MOBILESoft) emerged.

Mobile applications are as successful as ever, but there are strong signs that Mobile Software Engineering as a separate research field is coming to an end. In the following, we will take a closer look at the origins and development of Mobile Software Engineering, the current state of industry and research and the implications for future research.

2 MOBILE SOFTWARE ENGINEERING

The success of mobile applications had a strong impact on Software Engineering because mobile devices and their software have some differentiating features. According to Nunkesser and Thorn (2023) the most important ones are:

- Location independence
- Networking
- Context Sensitivity
- Simplicity
- Direct interaction

These features are not unique to mobile devices and applications, but they are more important for mobile devices and applications than for other devices and applications. Starting with these features, it is

[1] see https://www.apple.com/newsroom/2009/04/24Apples-Revolutionary-App-Store-Downloads-Top-One-Billion-in-Just-Nine-Months/

Nunkesser, R.
The End of Mobile Software Engineering (As We Know It).
DOI: 10.5220/0012780500003753

In *Proceedings of the 19th International Conference on Software Technologies (ICSOFT 2024)*, pages 131-136
ISBN: 978-989-758-706-1; ISSN: 2184-2833

possible to derive a number of challenges that are specific to mobile devices and applications. Wasserman (2010) lists the following challenges:

- Potential interaction with other applications
- Sensor handling
- Different development approaches (see Nunkesser (2018) for an overview)
- Families of hardware and software platforms (also referred to as *fragmentation*)
- Security
- Regulated User Interfaces
- Complexity of testing
- Power consumption

Many of these challenges require new research and development efforts. The research agenda proposed in Wasserman (2010) included the following areas and topics:

- User Experience
- Code reuse
- Networking
- Energy efficiency
- Data integrity
- Testing
- Portability and cross-platform development

Wasserman (2010) proposed the agenda as necessary "software engineering research related to [the] development of mobile applications". However, as Dehlinger and Dixon state in Dehlinger and Dixon (2011):

> [...] the growth of this new computing platform has outpaced the software engineering work tailored to mobile application development.

Therewith, they recognized the need for a separate research field, rather than just for research related to mobile applications.

As an initial example to confirm this need, let us consider a literature review from 2013. Salazar et al. (2013) focus on usability and state that "in spite of a common recognition of the importance of usability in the context of mobile devices, systematic research findings with respect to usability heuristics for mobile phones are sparse [...], this demonstrates clearly a lack of research in this area."

Let us consider Muccini et al. (2012) as a second example. The authors state that "mobile applications [are] (so) different from traditional ones", that they "require different and specialized new testing techniques" and therefore separate research is necessary.

The need for seperate research is also supported by needs from the industry. Joorabchi et al. (2013) provide a Grounded Theory based study on challenges in mobile software engineering. The study provides challenges that concern all areas of Wasserman (2010)'s research agenda except for energy efficiency. However, it is apparent that energy efficiency is an industry-relevant challenge that is of great importance for mobile devices. Thus, the study provides evidence that the research agenda proposed in Wasserman (2010) is relevant for the industry.

In the years following the rise of iOS and Android devices and the work of Wasserman (2010), the need for separate research in Mobile Software Engineering was recognized in the literature. Mobile Software Engineering may be seen as an effort to achieve the necessary pace to fill the above-mentioned research gaps. When we consider the International Conference on Software Engineering (ICSE) and the Symposium on the Foundations of Software Engineering (FSE), the two flagship software engineering conferences, we can see that the need for a separate research field was recognized at ICSE with the International Conference on Mobile Software Engineering and Systems (MOBILESoft)[2], which was first held in 2014, and at FSE with the International Workshop on Software Development Lifecycle for Mobile (DeMobile)[3], which was first held in 2013.

More papers directly addressing the relationship between Software Engineering and Mobile Software Engineering (or Software Engineering primarily for mobile devices) were published in the following years. Nagappan and Shihab (2016); Baresi et al. (2021) are two prominent examples that discuss the trends and challenges.

3 THE END OF MOBILE SOFTWARE ENGINEERING

Let us take a closer look at the relationship between ICSE and MOBILESoft to see if there are signs that Mobile Software Engineering as a separate research field is coming to an end.

Figure 1 shows the number of accepted research papers at ICSE and MOBILESoft from 2015–2023[4].

[2]MOBILESoft is held in conjunction with ICSE

[3]DeMobile was held in conjunction with FSE in 2013 and 2014

[4]In 2014, no data on accepted papers was published in the MOBILESoft conference proceedings.

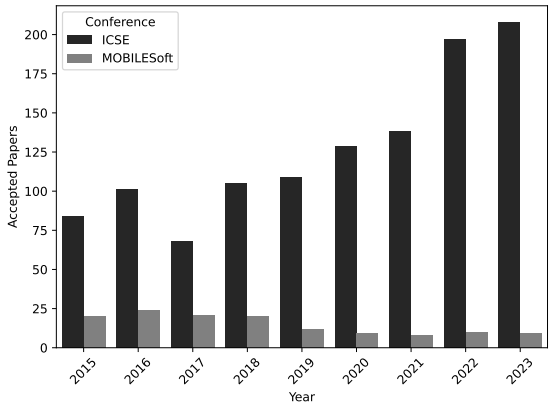

Figure 1: Accepted research papers at ICSE and MOBILESoft 2015–2023.

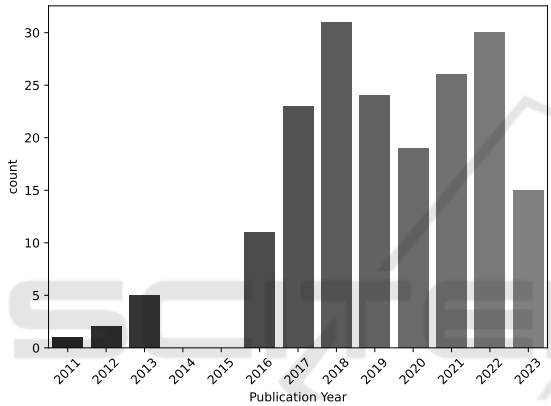

Figure 2: Publications from ICSE proceedings with mobile-specific keywords.

While ICSE shows a steady growth in publications, MOBILESoft's publications are declining - although the acceptance rate is higher than before (2023 had an acceptance rate of 47%; 2016 had an acceptance rate of 33%).

Does this trend mean that most of the research in Mobile Software Engineering is done? Of course not. It means that the need for separate research in Mobile Software Engineering is coming to an end and we are back to the need for "software engineering research related to the development of mobile applications" (Wasserman, 2010).

Figure 2 shows publications from ICSE proceedings where the keywords contained either "Mobile", "iOS", or "Android". The first publications were published in 2011 with a steady growth up to 2013. In 2014 - the year of the first MOBILESoft conference - and 2015, no such papers were published. However, mobile related papers appear again from 2016. It seems clear, at least for the considered conferences, that there was an initial boost for specific research in Mobile Software Engineering which came to an end.

In fact, it goes far beyond that: results and research that where obtained and conducted for mobile devices and applications have an impact on Software Engineering in its entirety.

4 MOBILE IS EATING THE WORLD

Andreessen (2011) stated that "Software is Eating the World". However, nowadays it is not an exaggeration to state that mobile is (also) eating the (software) world. According to StatCounter Global Stats Android first exceeded Windows in terms of internet usage in 2017 and in 2023 Android had a market share of 39.56% compared to 29.21% for Windows. Mobile is the dominant platform for internet usage and has to be considered in many software development projects.

4.1 Mobile First

From the perspective of the industry, there can be no doubt about the impact mobile has on Software Engineering. The term *Mobile First* is a good example for this observation. First used by Wroblewski (2011), the term described at the time the idea of designing web pages for all devices for mobile devices first. The importance of this paradigm change, which was adopted by Google in 2010, is self-explanatory. As a second example, let us consider Apple's strategy change with regard to macOS apps. A substantial reason for introducing technologies like SwiftUI and Mac Catalyst was to enable developers to do Mobile First for native development (see e.g. Sun, 2024, for the impact of iOS and iPadOS on macOS development). Examples of Apps developed *Mobile First* by Apple are News, Home, Stocks, and Voice Memos.

Similar observations can be made for other technologies that originated in mobile development and are now used in other areas. For example, the use of React Native for web and desktop development or the use of Flutter for web, desktop, and embedded development. .NET MAUI is another good example for a technology that originated in mobile development (Xamarin) and is now also used for desktop development.

4.2 Reconsidering the Research Agenda

The need for Mobile Software Engineering as a separate research field originated in the mobile software

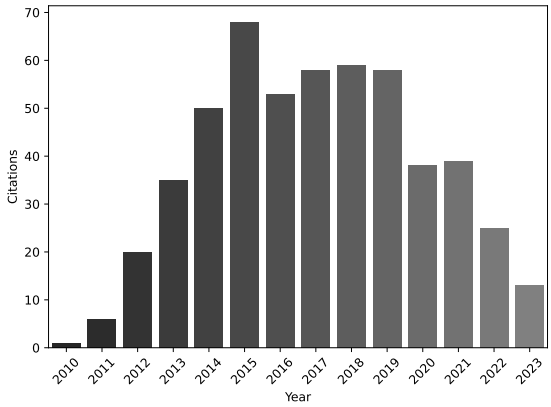

Figure 3: Citations of Wasserman (2010) by year according to ResearchGate.

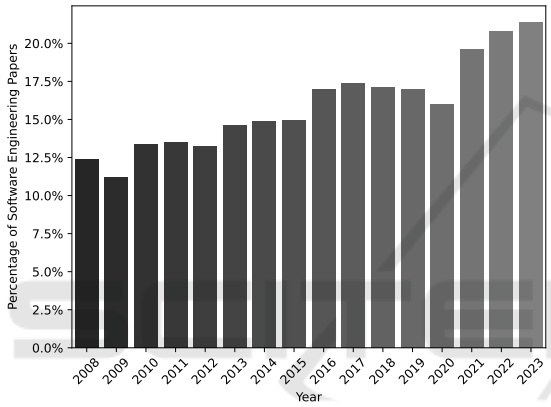

Figure 4: Percentage of papers with topics related to Wasserman (2010)'s research agenda according to Clarivate Web of Science.

boom starting in 2008, was recognized in the literature and most prominently addressed in the research agenda proposed in Wasserman (2010). In Section 3, we saw a research boom in Mobile Software Engineering from 2014–2019.

Figure 3 shows the number of citations of Wasserman (2010) by year according to ResearchGate. It shows a similar trend to the number of publications at MOBILESoft.

However, the declining citations and publications do not mean that the research agenda proposed in Wasserman (2010) is outdated. Figure 4 shows the percentage of papers with topics related to Wasserman (2010)'s research agenda according to Clarivate Web of Science, which is steadily increasing.

The research agenda proposed in Wasserman (2010) is now part of the research agenda of Software Engineering in its entirety. In 2023, only 11.64% of the papers in Figure 4 also cover mobile specific topics. Many topics that were originally mobile specific are now part of general Software Engineering research.

5 FUTURE PLANS

We have seen that the need for separate research in Mobile Software Engineering is coming to an end. However, the impact of mobile on Software Engineering is as strong as ever. Many papers still strongly distinguish between mobile and other devices at present, although the separation of research areas is no longer necessary.

The number of exceptions is limited. Beghoura et al. (2017), for example, consider energy consumption, inspire their research with the negative impact of high energy consumption, especially on mobile devices, and then proceed to develop a tool that is not limited to mobile devices. Maghawry et al. (2020) consider program transformation sequences to reduce lines of codes (LoC), motivated by limited resources on mobile devices, but this is also applicable to other devices. Oliveira et al. (2021) also consider energy consumption motivated by mobile development, but their research is not limited to mobile devices.

Future research should reconsider research conducted for mobile devices and applications and consider the impact on Software Engineering in its entirety. Researchers who are currently working on mobile-specific topics should also consider the impact of their research on Software Engineering. Conversely, Software Engineering researchers who have not included mobile devices and applications in their research so far on the other hand, should consider the impact of mobile on their research.

With this approach, Software Engineering will benefit from the generalization of previously overlooked research results. The aforementioned research topic of energy consumption is a good example. Manotas et al. (2016) show the relevance of energy consumption for the entirety of Software Engineering by stating that

> [...] 40 % of experienced traditional practitioners indicated that they have energy requirements or goals.

However, relevant and generalizable research like Cruz and Abreu (2019) only discuss the applicability of their results to "Cyber-Physical Systems and Internet of Things". This seems too limited given the observation that of the 22 discussed design patterns only "WiFi over Cellular" and "SensorFusion" are strictly mobile specific and many others are also relevant for saving energy on other devices and applications.

Mobile application development will also benefit from Software Engineering research results that did not consider mobile devices and applications before. Nunkesser. (2022) for example, consider the application of Hexagonal Architecture for mobile application development.

As a final note, it is certainly also worth looking at related research fields such as Embedded Software Engineering, Internet of Things and Cyber-Physical Systems to see if there are research results that can be generalized to Software Engineering.

6 CONCLUSION

In this paper, we have seen that Mobile Software Engineering as a separate research field has been relevant in accelerating research in the area of mobile devices and applications. However, the need for separate research in Mobile Software Engineering is coming to an end. Researchers who up until now have concentrated on mobile specific topics should elicit the possibilities of generalizing their research. Researchers who have not included mobile devices and applications in their research so far should consider the impact of mobile on their research.

It's the end of Mobile Software Engineering as we know it and I feel fine.

REFERENCES

Marc Andreessen. 2011. Why Software Is Eating The World. *The Wall Street Journal* August 20, 2011 (2011). http://online.wsj.com/article/SB10001424053111903480904576512250915629460.html

Luciano Baresi, William G. Griswold, Grace A. Lewis, Marco Autili, Ivano Malavolta, and Christine Julien. 2021. Trends and Challenges for Software Engineering in the Mobile Domain. *IEEE Softw.* 38, 1 (jan 2021), 88–96. https://doi.org/10.1109/MS.2020.2994306

Mohamed Amine Beghoura, Abdelhak Boubetra, and Abdallah Boukerram. 2017. Green software requirements and measurement: random decision forests-based software energy consumption profiling. *Requirements Engineering* 22, 1 (March 2017), 27–40. https://doi.org/10.1007/s00766-015-0234-2

Luis Cruz and Rui Abreu. 2019. Catalog of energy patterns for mobile applications. *Empirical Software Engineering* 24, 4 (Aug. 2019), 2209–2235. https://doi.org/10.1007/s10664-019-09682-0

Josh Dehlinger and Jeremy Dixon. 2011. Mobile application software engineering: Challenges and research directions. *Workshop on mobile software engineering* 2, 29–32.

Mona Erfani Joorabchi, Ali Mesbah, and Philippe Kruchten. 2013. Real Challenges in Mobile App Development. In *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*. 15–24. https://doi.org/10.1109/ESEM.2013.9

Ahmed Maghawry, Mohamed Kholief, Yasser Omar, and Rania Hodhod. 2020. An Approach for Evolving Transformation Sequences Using Hybrid Genetic Algorithms. *International Journal of Computational Intelligence Systems* 13 (2020), 223–233. Issue 1. https://doi.org/10.2991/ijcis.d.200214.001

Irene Manotas, Christian Bird, Rui Zhang, David Shepherd, Ciera Jaspan, Caitlin Sadowski, Lori Pollock, and James Clause. 2016. An empirical study of practitioners' perspectives on green software engineering. In *Proceedings of the 38th International Conference on Software Engineering* (Austin, Texas) *(ICSE '16)*. Association for Computing Machinery, New York, NY, USA, 237–248. https://doi.org/10.1145/2884781.2884810

Henry Muccini, Antonio Di Francesco, and Patrizio Esposito. 2012. Software testing of mobile applications: Challenges and future research directions. In *2012 7th International Workshop on Automation of Software Test (AST)*. 29–35. https://doi.org/10.1109/IWAST.2012.6228987

Meiyappan Nagappan and Emad Shihab. 2016. Future Trends in Software Engineering Research for Mobile Apps. In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, Vol. 5. 21–32. https://doi.org/10.1109/SANER.2016.88

Robin Nunkesser. 2018. Beyond web/native/hybrid: a new taxonomy for mobile app development. In *Proceedings of the 5th International Conference on Mobile Software Engineering and Systems* (Gothenburg, Sweden) *(MOBILESoft '18)*. Association for Computing Machinery, New York, NY, USA, 214–218. https://doi.org/10.1145/3197231.3197260

Robin Nunkesser. 2022. Using Hexagonal Architecture for Mobile Applications. In *Proceedings of the 17th International Conference on Software Technologies - ICSOFT*. INSTICC, SciTePress, 113–120. https://doi.org/10.5220/0011075100003266

Robin Nunkesser and Jens Thorn. 2023. Possibilities and Limitations of Mobile Applications for Controlling. In *The Digitalization of Management Accounting: Use Cases from Theory and Practice*, Imke Keimer and Ulrich Egle (Eds.). Springer Fachmedien Wiesbaden, Wiesbaden, 243–261. https://doi.org/10.1007/978-3-658-41524-2_16

Wellington Oliveira, Renato Oliveira, Fernando Castor, Gustavo Pinto, and João Paulo Fernandes. 2021. Improving energy-efficiency by recommending Java collections. *Empirical Software Engineering* 26, 3 (April 2021), 55. https://doi.org/10.1007/s10664-021-09950-y

Luiz Henrique A Salazar, Thaísa Lacerda, Juliane Vargas Nunes, and Christiane Gresse von Wangenheim. 2013. A Systematic Literature Review on Usability

Heuristics for Mobile Phones. *International Journal of Mobile Human Computer Interaction (IJMHCI)* 5, 2 (2013), 50–61. https://doi.org/10.4018/jmhci.2013040103

Haoxuan Sun. 2024. The Smartphone Revolution: A Comparative Study of Apple and Samsung. *Highlights in Business, Economics and Management* 24 (Jan. 2024), 575–580. https://doi.org/10.54097/5q0mnb51

Anthony Wasserman. 2010. Software engineering issues for mobile application development. *Proceedings of the FSE/SDP Workshop on the Future of Software Engineering Research, FoSER 2010*, 397–400. https://doi.org/10.1145/1882362.1882443

Luke Wroblewski. 2011. *Mobile First*. A Book Apart.