GLL

PERFORMANCE OF MAP APPLICATIONS ON MOBILE DEVICES: CASE STUDY

Karol Król, Jozef Halva, Alexandra Pagáč Mokrá

Summary

Performance is one of the most important indicators of quality of software. It plays a significant role not only for business websites and apps, but in case of map components and applications, especially for mobile devices. The aim of the paper is to measure performance of map applications on mobile devices. The performance testing was run on a prototype of an interactive map component of the zoom-lens type using selected web applications. Speed Index (measurement unit) reached the value of 1601 milliseconds, and Performance Score obtained the highest possible value (100 units). A high value of Lighthouse Performance Score was also noted (98 units). Study showed that satisfying measurement results of indices in relation to the time of loading of the component in the browser window are associated with a small size of the component. In case of small components, which enhance the functionality of map websites and apps, it is recommended to compress all parts of the component, regardless of the results of the performance tests.

Keywords

website performance • website speed test • loading speed • conversion rate • quality indices

1. Introduction

Performance testing is one of the indices of quality of software (web, mobile, etc.), and sometimes is a point of reference for further development. Software performance is dependent on various factors, such as response, speed and underling resources [Patel and Gulati 2015]. Although performance testing involves non-functional testing, it is relatively often done, and there are many, who decide to run it. Currently, before releasing, enterprises usually conduct full performance testing of their applications to ensure that the application responds fast enough [Agile 2012].

Software developers typically measure a web application's quality of service in terms of response time, webpage availability, and throughput. Performance testing and evaluation of software components becomes a critical task. Poor quality of software performance can lead to missing opportunities [Patel and Gulati 2014, p. 1297].
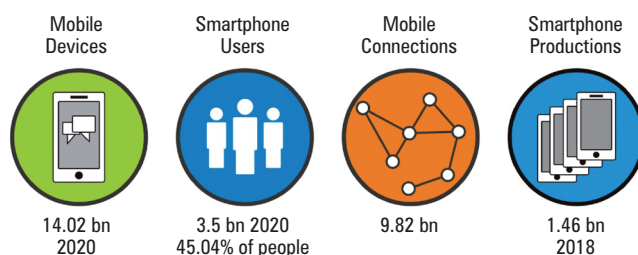
Performance is also important for map components and applications, on mobile devices in particular [Król 2018a]. Applications designed on database technologies

have an alternative in form of raster components, which despite their limitations can be used for displaying spatial data [Król and Szomorova 2015, Król and Prus 2016, Król and Bitner 2019]. Image viewer performance can have a significant impact on user experience. Raster maps browsers can be used by various entities, including public administration institutions. They are useful when there is no need to create complex map websites or to quickly publish a map [Król 2018b]. The aim of the paper is to measure the performance of an map application on mobile devices.

The paper is organised as follows. The second section focuses on the increasing number of users of mobile devices. In the next section, it is described how website performance (map application) translates into target conversion. The „materials and methods" section presents the research assumptions, testing applications and the tested component. Subsequent section provides with the results of the study and their discussion. The entire paper is concluded with a summary.

2. Mobile technology facts in the world

About 1.46 billion smartphones were manufactured in 2018. The number of mobile devices is expected to grow from 14 billion in 2020 to 16.8 billion in 2023. The number of global smartphone users increased by 40% in 2016–2020. Today, it exceeds three billion and will increase by several hundred million in the nearest future. The number of smartphone users grew from 2.5 billion in 2016 to about 3.5 billion in 2020 (Fig. 1). Statista predicts substantial growth for all cellular-enabled phones, tablets, or IoT devices on a global scale. The company's data sources estimate that in the next three years, we will see the number of mobile device users increase to 7.33 billion [Statista 2020].

Mobile Devices	Smartphone Users	Mobile Connections	Smartphone Productions
14.02 bn 2020	3.5 bn 2020 45.04% of people	9.82 bn	1.46 bn 2018

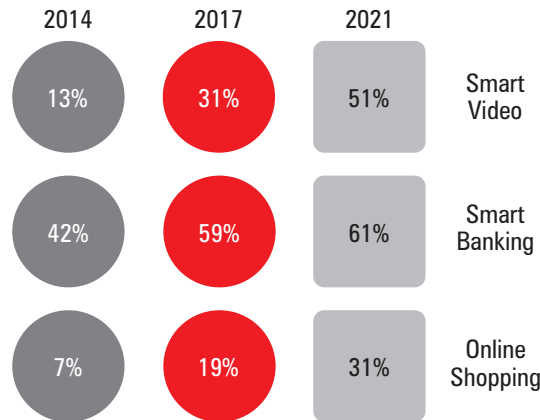Source: Authors' own study based on Statista 2020

Fig. 1. The global popularity of smartphones

The global smartphone sales in 2017 was USD 478.7 billion. According to GSMA real-time intelligence data, there are now over 9.82 billion mobile connections worldwide, which surpasses the current world population of 7.77 billion indicated by UN digital analyst estimates. This data mean there are about 2 billion more mobile connections than people worldwide [Turner 2020].

The economic importance of mobile technologies grows as fast as their social impact. In 2015, Poles purchased PLN 2.5 billion worth of goods and services through mobile channels and sent 559 million gigabytes of data, 114% more than the year before. The year 2018 was the first when a greater number of people using mobile devices than computers for online activities [IAB Polska 2018, IAB Polska 2019].

In 2018, 90% of online users between 15–24 years of age have accessed the Internet through smartphones. For the 25–34 group, this was almost 95%. For senior users (55+) – about 80%. The 'mobile consumers of the future' are mostly young people aged 18 to 24 who represent the so-called 'mobile generation'. As many as 80% of them use mobile devices all the time [IAB Polska 2018].

Smartphone users most often use navigation and mobile applications, but online activities such as e-banking, e-shopping, online video, and browsing gain in popularity [Jeong et al. 2016]. Smartphone use is forecasted to grow mostly for payments, mobile banking, and online shopping (Fig. 2).

Source: Authors' own study based on IAB Polska [2018]

Fig. 2. IAB Polska Mobile, Internet users 15+, smartphone users

The increased number of users of smartphones and other online devices changed the website and web application design philosophy. The traditional solutions were simply not enough. Static websites with an a priori set width became dysfunctional. A need arose for websites and applications that are convenient to use regardless of the device size. It applies to public administration websites as well.

3. The need for performance testing

More than ever, website's performance matters, and the pressure put on performance keeps growing. Thus, the need for performance testing is growing, and many companies are moving toward automated performance testing. This follows the expectations

of Internet users, who do not tolerate slow websites and web applications. Moreover, every second of delay may have a negative impact on target conversion and rank position in browsers. The average Internet user expects websites to load within two seconds or faster [Gomez 2010].

Total size of websites has been increasing steadily since at least 2011. Despite insistence on files compression and code minification this trend seems to continue [Wager 2020]. When the website's performance is low, clients tend to order less frequently and impedes the retailers' desire to turn online traffic into sales [Gallino et al. 2018]. According to a survey by Aberdeen Group, up to 46% enterprises do not have instruments to monitor web applications performance. Moreover, Aberdeen Group has found that a one second delay in web page response time led to a decrease of the conversions by 7% [Arsenault 2015]. Pinterest has increased search engine traffic and sign-ups by 15% when they reduced the perceived waiting time by 40%. COOK has increased conversions by 7%, decreased bounce rates by 7%, and increased pages per session by 10% when they reduced the average page load time by 850 milliseconds. BBC has found it lost 10% of users for every additional second it took for its site to load. DoubleClick by Google has found that 53% of mobile site visits were given up if a page took longer than 3 seconds to load [Wagner 2020].

4. Materials and methods

Performance testing is a procedure performed to determine the speed and the effectiveness of a system. It basically focuses on determining whether a user of the system will be satisfied with the performance characteristics of the application. Performance optimization (also called performance tuning) is a procedure carried out to improve the product's quality and meet the users' expectations [Patel and Gulati 2018].

Webpage content, architecture, and the use of content delivery networks are key components that determine website performance [Gallino et al. 2018]. Therefore, performance tuning can sometimes be more of an art than science, due to the sheer complexity of the systems involved in modern web application design. A performance test can help to determine whether the product meets the performance goals or not [Patel and Gulati 2014].

The main focus of performance testing of any website depends on its: (1) Speed – determines whether the application responds quickly; (2) Scalability – determines maximum user load the software application can handle, (3) Reliability – determines the readiness of the system, and (3) Stability – determines if the application is stable under varying loads [Agile 2012, Patel and Gulati 2014]. 'Performance Testing' is a generic term which is used to describe a category or suite of tests. Performance testing can be of many types, including: Load test, Stress test, Soak/Endurance Testing, Volume Testing, and Scalability test (Fig. 3). It can also by conducted from various perspectives: (1) developers, (2) files volume (size, quantity and the type of resources), (3) testing organisation, (4) type of performance testing and (5) used tools [Leitner and Bezemer 2017].
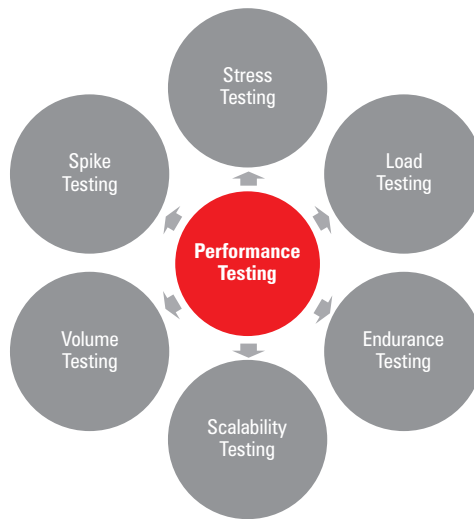
Source: developed by Authors' based on Agile [2012]

Fig. 3. Types of performance testing

Performance testing was run on a prototype of an interactive map component of the zoom-lens type. The component allows to browse zoomed raster files (Fig. 4). It is built with the use of jQuery scripts (TikslusZoom v 1.1.0). Tikslus zoom is a jquery based image zoom plugin which can be easily customised.
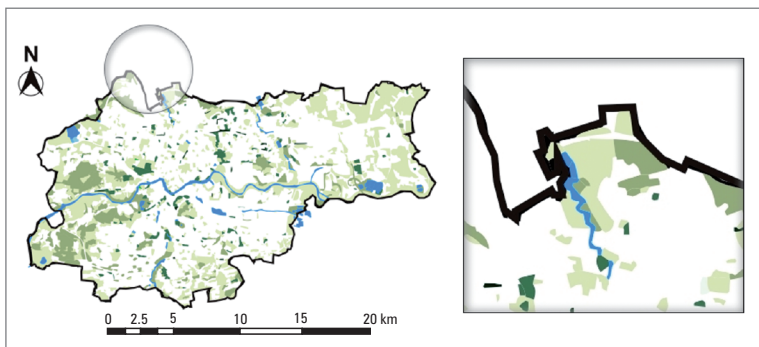
Source: Authors' own study

Fig. 4. View of the zoom-lens type map component (print screen)

The performance measurements include various applications that automatise the testing [Maila-Maila et al. 2019]. Individual tests were the pillar of the research, i.e. ad-hoc verification testing. Such tests are performed for preliminary control, allowing to evaluate the web app's performance in the initial phase (before releasing the application).

Testing is run on selected tools (Table 1), in mobile mode, in order to evaluate the initial performance of the map component on portable devices. The testing applications were chosen according to availability criterion. Each of them is free of charge and runs the tests in a web browser window, thus it is not necessary to install additional software.

Table 1. Online tools used in the performance tests

ID	Performance testing tool	Key performance indicators (range or measurement unit)
1.	Geekflare	Performance Score (0–100)
		Loaded In (s)
		First Byte In (ms)
		Page Size (MB)
2.	mobiReady	mobiReady Score (0–5)
3.	Dareboost	Weight (KB)
		First Byte (s)
		Fully Loaded (s)
		Speed Index (> 0)
4.	PageSpeed Insights	First rendered content (s)
		Speed index (s)
		Lighthouse Performance score (0–100)
5.	Lighthouse	Time to Interactive (s)
		Largest Contentful Paint (s)

Explanations:
1. https://gf.dev/website-audit
2. https://ready.mobi/
3. https://www.dareboost.com
4. https://developers.google.com/speed/pagespeed/insights/
5. Chrome DevTools

Source: Authors' own study

Each of testing applications measures different performance indices, which guarantees a kind of 'complementarity testing'. Selected performance indices are related to the speed of application loading in a web browser window [Król i Zdonek 2020]. Other indices are synthetic and aggregated, such as Performance Score (Geekflare), Speed Index (Dareboost) or Lighthouse Performance score (PageSpeed Insights). Lighthouse (Chrome browser extension) is an open-source, automated tool for improving the performance, quality, and correctness of your web apps. Lighthouse was used to meas-

ure the performance of Time to Interactive (s) and Largest Contentful Paint (s). Time to interactive is the amount of time it takes for a page to reach full interactivity. Largest Contentful Paint marks the time at which the largest text or image is painted.

5. Study results and discussion

The tested component had relatively small total size that did not exceed one megabyte (about 900 KB). Its size was determined mainly by the raster files size (about 750 KB). For this reason, performance tests showed maximum possible value for aggregated indices Performance Score and Lighthouse Performance Score, but also satisfying measurement results of indices related to time of component loading in browser window was noted (Table 2). However, their values could be even better. Raster compression could improve the component's performance. Król and Bitner [2019] have shown that for applications based on raster maps, which are view-only and temporary (ad-hoc applications), maintaining (very) high quality of graphic files at the expense of application performance is not necessary.

Table 2. Performance testing results

Performance testing tool	Key performance indicators (range or measurement unit)	Measurement value
Geekflare	Performance Score (0–100)	100
	Loaded In (s)	0.7
	First Byte In (ms)	118
	Page Size (MB)	0.2
mobiReady	mobiReady Score (0–5)	3
Dareboost	Weight (KB)	0.18
	First Byte (s)	0.41
	Fully Loaded (s)	2.61
	Speed Index (ms)	1601
PageSpeed Insights	First Byte (s)	1.7
	Speed Index (s)	1.7
	Lighthouse Performance score (0–100)	98
Lighthouse	Time to Interactive (s)	1.6
	Largest Contentful Paint (s)	2.4

Source: Authors' own study

Speed Index (measurement unit) reached a value of 1601 milliseconds, which is relatively good, although slightly too high for such a small component. This is due to a large size of the raster file, which was a 'static' resource in this case. Speed Index is a destimulant, which means that lower values are desirable. The faster the rendering (higher performance), the lower the index value, with Google recommending that it should not exceed 1000 units [Król 2018].

The 'weakest' link of the tested component is the raster file, which is displayed in the extended window. Loading of the resource is felt, although it takes a fraction of a second. However, this delay is smoothed by an animation. Research show that loading animations, such as progress bars and spinners, inform users of the current working state and make the process more tolerable to the user. Users are willing to wait longer when the site uses a dynamic progress indicator [Sherwin 2014].

Measurements taken with the Lighthouse tool showed that the waiting time of the main 'component' of the application was relatively long – 2.4 seconds. The tested component reached the 'full interactivity' in browser window slightly faster. These are, however, only results of algorithmic measurements. The component is fully functional only when all its resources are loaded.

6. Summary

The results obtained are synthetic, 'mechanical' and illustrative. Algorithmic tests should be supplemented by usability test run under normal conditions of use, as when using smartphones, and in extreme conditions (stress testing). Satisfying performance of the component is not the only feature that determines the usability of web application.

In the case of small components, which enhance website and map application functionality, it is advisable to compress all elements of the component, such as scripts, graphic files, programming libraries, cascade style sheets etc. It is also recommended to consolidate the resources and limit or completely eliminate components retrieved from external sources as they may delay the loading of the component in a browser window.

References

Agile 2012. Application Performance Testing Basics. Agileload Blog. https://bit.ly/performance-testing-basics [accessed: 20.06.2020].

Arsenault R. 2015. Slow is the New Downtime When it Comes to Web Performance. Aberdeen Group. https://bit.ly/Aberdeen-G [accessed: 20.06.2020].

Gallino S., Karacaoglu N., Moreno A. 2018. Why Retailers Should Care about Net Neutrality: The Impact of Website Performance on Online Retail. http://dx.doi.org/10.2139/ssrn.3260203

Gomez 2010. Why Web Performance Matters: Is Your Site Driving Customers Away? White Paper. https://bit.ly/Web-Performance-Matters [accessed: 20.06.2020].

IAB Polska 2018. Mobile 2018. IAB Polska. https://iab.org.pl/badania-i-publikacje/raport-mobile-2018-2/ [accessed: 20.06.2020].

IAB Polska 2019. Internet 2018/2019. IAB Polska. https://iab.org.pl/wp-content/uploads/2019/06/HBRP-raport-IAB-05-191.pdf/ [accessed: 20.06.2020].

Jeong S.H., Kim H., Yum J.Y., Hwang Y. 2016. What type of content are smartphone users addicted to?: SNS vs. games. Computers in Human Behavior, 54, 10–17. https://doi.org/10.1016/j.chb.2015.07.035

Król K. 2018a. Performance threshold of the interactive raster map presentation – as illustrated with the example of the jQuery Java Script component. Geographic Information Systems Conference and Exhibition "GIS Odyssey" 2018, 321–327.

Król K. 2018b. Comparative analysis of the performance of selected raster map viewers. Geomatics, Landmanagement and Landscape (GLL), 2, 23–32. https://doi.org/10.15576/GLL/2018.2.23

Król K., Bitner A. 2019. Impact of raster compression on the performance of a map application. Geomatics, Landmanagement and Landscape (GLL), 3, 41–51. https://doi.org/10.15576/GLL/2019.3.41

Król K., Prus B. 2016. The comparative analysis of selected interactive data presentation techniques on the example of the land use structure in the commune of Tomice. Polish Cartographical Review, 48 (3), 115–127. https://doi.org/10.1515/pcr-2016-0009

Król K., Szomorova L. 2015. The possibilities of using chosen jQuery JavaScript components in creating interactive maps. Geomatics, Landmanagement and Landscape (GLL), 2, 45–54. https://doi.org/10.15576/gll/2015.2.45

Król K., Zdonek D. 2020. Aggregated Indices in Website Quality Assessment. Future Internet, 12 (4), 72. https://doi.org/10.3390/fi12040072

Leitner P., Bezemer C.P. 2017. An exploratory study of the state of practice of performance testing in java-based open source projects. In: Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering, 373–384. https://doi.org/10.1145/3030207.3030213

Maila-Maila F., Intriago-Pazmiño M., Ibarra-Fiallo J. 2019. Evaluation of Open Source Software for Testing Performance of Web Applications. In: New Knowledge in Information Systems and Technologies. WorldCIST'19. Advances in Intelligent Systems and Computing, eds. Á. Rocha, H. Adeli, L. Reis, S. Costanzo, 931, 75–82, Springer, Cham. https://doi.org/10.1007/978-3-030-16184-2_8

Patel C., Gulati R. 2014. Software Performance Testing Measures. International Journal of Management & Information Technology, 8 (2), 1297–1300. https://doi.org/10.24297/ijmit.v8i2.681

Patel C., Gulati R. 2015. Identifying ideal values of parameters for software performance testing. In: 2015 International Conference on Computing, Communication and Security (ICCCS), 1–5. IEEE. https://doi.org/10.1109/CCCS.2015.7374125

Patel C., Gulati R. 2018. WebN: A Strainer Approach for Website Performance Analysis. In: Networking Communication and Data Knowledge Engineering. Lecture Notes on Data Engineering and Communications Technologies, eds. G. Perez, K. Mishra, S. Tiwari, M. Trivedi, 4, 3–13. Springer, Singapore. https://doi.org/10.1007/978-981-10-4600-1_1

Sherwin K. 2014. Progress Indicators Make a Slow System Less Insufferable. Nielsen Norman Group. https://www.nngroup.com/articles/progress-indicators/ [accessed: 20.06.2020].

Statista 2020. Forecast number of mobile devices worldwide from 2019 to 2023 (in billions). https://bit.ly/3aF0vYM [accessed: 20.06.2020].

Turner A. 2020. How Many Mobile Connections Are There Worldwide?. https://bit.ly/2AUFCNi [accessed: 20.06.2020].

Wagner J. 2020. Why Performance Matters. Web Fundamentals. Google Developers. https://bit.ly/Performance-M [accessed: 20.06.2020].

Dr inż. Karol Król
Uniwersytet Rolniczy w Krakowie
Katedra Gospodarki Przestrzennej i Architektury Krajobrazu
ul. Balicka 253c, 30-149 Kraków
http://digitalheritage.pl
ORCID: https://orcid.org/0000-0003-0534-8471
e-mail:k.krol@onet.com.pl

Ing. Jozef Halva, PhD.
Slovak University of Agriculture in Nitra
Department of Landscape Planning and Land Consolidation
Hospodárska 7, Nitra, Slovakia
e-mail: jozef.halva@uniag.sk

Ing. Alexandra Pagáč Mokrá
Slovak University of Agriculture in Nitra
Department of Landscape Planning and Land Consolidation
Hospodárska 7, Nitra, Slovakia
e-mail: xmokraa@is.uniag.sk