

# Abstract—nowadays, of these apps called mobile web

[Design](#)



Abstract—Nowadays, a large set of mobile applications are provided in several types, one of these apps called mobile web application which standardized by World Wide Web Consortium (W3C), and based on web app manifest that gives the ability for developers to include metadata information about the web application into JSON datatype file.

This metadata provides the developers to declare properties that control the behavior of web application in the device and add restrictions, for example, specified a labeled icon for web application when application added to home screen. This research collected a large dataset of mobile web applications that have JSON metadata file, these metadata files have been analyzed to extract some statistical information about the features and properties that used to provide mobile web application. Also, this research analyzed some of the sub-properties into metadata such as (display, background\_color, icons, lang, orientation, permissions, and theme\_color).  
Keywords—Android, Mobile, Web, Applications, Manifest, Analysis. I.

Introduction Mobile applications in the Android platform can be delivered in two ways: as client-side apps or as web apps. In the client-side apps, an app is installed on the user devices in an Android Package Kit (APK) and developed using the Android Software Development Kit (SDK), in another hand, web applications are client-server computer programs in which a client, including client-side logic and the user interface, runs using embedded web browser [1]. There are common examples for web applications such as webmail, online auctions, online retail sales, instant messaging services, wikis and others [2]. The main distinction between a web application and a dynamic web page of any kind is unclear, however, websites are most likely to

<https://assignbuster.com/abstractnowadays-of-these-apps-called-mobile-web/>

be indicated as web applications which have similar functionality to a mobile app, or to a desktop software application<sup>2</sup>. In a web application, there are several ways of targeting mobile devices in the mobile web: responsive web design and progressive web apps.

Responsive web design used for making a web application, mobile apps or native apps to directly run on a mobile device. In another hand, progressive web apps are a hybridization of a mobile application and regular web pages (websites), in which a mobile website is embedded inside a native app<sup>3 4</sup>. The web app manifest contains metadata about an application in a JSON datatype file such as (name, theme\_color, icon, permissions). The main goal of the manifest file is to store web applications to the home screen of a device as a site bookmark and act like installable application, also giving users with more wealthy experience and more quick access<sup>5</sup>.

web app manifest gives the responsibility of the developer for controlling the application appears to the user in regions that they expect for watching an app, guiding what they can launch, and determine the appearance at launch. When a site is launched it has a unique name and icon which users can differentiate it from other sites, it shows for user something while resources are restored or downloaded from cache, it gives default display features to the browser to prevent abrupt transition when site available and it does all this over the simple techniques of metadata in a text file, which called the web app manifest<sup>5</sup>. Web technologies involve web app manifest as progressive web apps, that can be downloaded to the device without using the application store, in addition to other features like getting

push notifications and being available when offline. In this paper, our target is to analysis the manifest. jsonfile for a large dataset of web apps to extract some useful statistical information for a set of selected properties. The rest of the paper is organized as follow: section 2 shows some of related works, section 3 describes the steps performed to collect and process our dataset, section 4 the experiments and results and a brief discussion of them, finally a conclusion and proper future work are represented in section 5. II.

Related work (Charland et. al, 2011)<sup>6</sup>, discussed the weakness and compared between the two types of apps in term of code, user interface code, user experience, performance, and design. (Henning Heitkötter et. al 2013) <sup>7</sup> developed eleven criteria to evaluate mobile web frameworks, the need of developing these criteria is due to the difficulty of selecting an appropriate framework among a plenty of them. The authors classified the proposed criteria into two categories; developers and user's perspective. Seven of the proposed criteria are from a developer perspective: License and Costs, Long-term Feasibility, Documentation and Support, Learning Success, Development Effort, Extensibility, and Maintainability, and four of them are from user's perspective: User Interface Elements, Native Look & Feel, Load Time, and Runtime Performance.

The authors then used these criteria to evaluate four mobile web frameworks: jQuery Mobile, Sencha Touch, The-M-Project, and Google Web Toolkit. The results showed that jQuery Mobile is the best for mobile UI, while Sencha Touch is suitable for mobile web applications that have high complexity. (Serrano et. al, 2013)<sup>8</sup>, examined the different current

approaches for developing mobile web apps to help developers making a correct decision when they need to select an approach, they classified these approaches into five types: standard web apps, responsive web apps, mobile web apps, hybrid apps, and native apps. The authors then provide a set of technical and non-technical criteria which can be used by the developers to help them decide which type of the above-mentioned approaches will be more suitable to their situation than the others. The technical and non-technical criteria. The technical criteria are platforms and version support, device capabilities, user experience, performance, and upgrade.

On another hand, the non-technical criteria are distribution, approval cycle, and monetization. (Ivano Malavolta et. al, 2016) 9, describe the content of a tutorial for web-based hybrid mobile applications. The tutorial discusses the problems occurred when the development and maintenance of the mobile apps are platform dependent (i. e. each platform using different programming language and tools). Web-based hybrid mobile apps are the solution proposed by well-known companies such as IBM and Adobe.

The proposed solution enables the developers to use standardized web technologies and then use cross-platform wrappers and other tools to spread them in app stores. Many benefits will be gained by using the proposed solution such as the ability of use cross-platform, exploitation of the existing knowledge of the developers, simpler, and less expensive. However, on another hand, there are some drawbacks such as constrained access to the hardware, the differences between user's experiences, and decreasing in performance. (Sheppard et. al, 2017) 10, provide a good tutorial to how you

can add an app to your mobile home screen and how you can control it using web app manifest. The authors then provide a description of the properties which can be controlled using manifest. json file. Because of following the tutorial, a web app will be as same as possible to its corresponding native app; it will have launched faster, work as offline apps, loaded faster, and will be launch from the home screen.

III. Dataset The dataset has been collocated as domain names from “publicwww” [11], which a website that has a search engine used to search into source code (HTML) of websites over internet, to cover a big dataset we have used this query keyword “ ” which return the domain names that have “manifest. json” file in their root directory into a CSV file and we have collocated 19222 domains. After that the domain names in CSV file have been preprocessed into MySQL database using our “read. php” script which store the dataset into programmer friendly style, the MySQL database has three columns (id, url, rank), the URL contains the domain name, and rank contains the domain rank over the web from CSV file.

Then the list of domain names has been crawled using “crawler. php” script which downloads “manifest. json” files and store it into dataset directory with their id value from the database as a filename. The dataset has been reviewed and removed bad results that do not have normal “manifest. json” file structure and the total dataset for the experiment was 14784 files.

IV. EXPERIMENTS and results After collecting and processing dataset the “json. php” file decoded the dataset JSON files into Array datatype, the code algorithm count Array keys (property of JSON file) and values for each JSON

<https://assignbuster.com/abstractnowadays-of-these-apps-called-mobile-web/>

file, the result of first iteration that counts the JSON properties from Arrays we found that 62810 main property in all JSON files which can be seen in figure 1, properties less than 200 have been excluded from figure 1 which have (y-axis shows properties names, x-axis have the count of each property and each column have the percent of property in dataset), we notice that most of the mobile web applications have “ name” property and 15.3% doesn't have icons the rest of results displayed in the figure and the whole table can be seen in Appendix Table 1. Figure 1 Main properties After counting primary properties, we have selected some of these properties to be analyzed: A- display: the display property has three main types: I. standalone: that make application act like a native application. II.

browser: make the application act like a bookmark of the webpage. III. fullscreen: make the ability to launch the application on the whole screen. IV. minimal-ui: like full screen but with less content in the user interface and minimum navigation buttons.

The results were after the analysis: (standalone: 96.03%, browser: 2.38%, fullscreen: 1.05%, minimal-ui: 0.54%). B- orientation: in this property restrict the application to specific orientation the results were: (portrait: 69%, natural: 12.4%, any: 8.

6%, landscape: 6.4%, portrait-primary: 3.4%, landscape-primary: 0.

2%), which indicate that the most of mobile web applications use portrait orientation mode. C- lang: this property declares the language of application in ISO representation, the most language with their localization of

applications was English around 47% of the dataset. D- permission: this property identifies the special permissions that the application need from the device, the most applications around 58.45% uses “ gcm” (Google CloudMessaging) permission, 19.2% uses notifications permission and 14.61% use the storage of the device, the rest permissions shown in Table 1.

Permission Percentage gcm 58.45% notifications 19.20% storage 14.61% pushMessaging 2.29% webRequest 0.

57% webNavigation 0.29% background 0.29% tabs 0.29% cookies 0.

29% webRequestBlocking 0.29% activeTab 0.29% Table 1

Permissions E- theme\_color: this property serves default color of the theme in the context of the application, the result for over 1% was (62.44% for white color, 2.61% for black color and 1.61% for steel blue color).

F- background\_color: this property describes the predictable color of background in application the result for over 1% was (64.5% for white color, 4.1% for gray color, 2.2% for light steel blue color and 1.7% for black

color) G- icons: this property represents iconic of application in mobile with three sub-properties (src: which identify image file location and all images have src, sizes: describe the icon size, type: classify icon meta-type). Size Percentage 192×192 21.

0% 144×144 14.1% 96×96 13.4% 72×72 12.9% 48×48 12.

9% 36×36 12.2% 512×512 4.4% 256×256 3.1% 384×384 1.2% Table 2 icons sizes The results for sizes for sizes that bigger than 1% in table 2 and



for icon meta type the result was (“ image/png” has 99. 2%, “ image/jpeg” has 0. 21%, and other types have percentage under 0.

2%). V. Conclusion and future work In this work, we discussed the statistical analysis of metadata mobile web application files for a large dataset. the results indicate that a lot of applications doesnot use all the properties of metadata and a lot of applications have missed important properties such as icons that have 84. 6% of overall dataset. In future, we plan to study more properties indeep using data mining models to extract knowledge about metadata manifest.