

A Systematic Mapping Study of Cross-Platform Mobile Apps

¹Tasnim Zohud and ²Samer Zein

Department of Computer Science, Birzeit University, Palestine

Article history

Received: 26-12-2018

Revised: 18-03-2019

Accepted: 23-04-2019

Corresponding Author:

Tasnim Zohud

Department of Computer
Science, Birzeit University,
Palestine

Email: t.zuhod@gmail.com

Abstract: Cross-platform frameworks for mobile Application (app) development allow developers to deploy solutions on a range of platforms such as iOS and Android. With the increased use of such frameworks, it is of utmost importance to understand the contributions and challenges in this emergent field. Although there have been many studies in this area of research, there is a lack of a coherent view. To address this issue, we have conducted a Systematic Mapping Study (SMS) to categorize the available studies in the area of cross-platform mobile app development. More specifically, we aim to understand the accomplishments, contribution facets and the challenges presented in this field. Thirty (30) studies are mapped to a classification scheme. Different research gaps are identified: more research needs to be conducted using more complex and real-world apps; the point of how cross-platform apps are identified and distinguished from the native is still not clarified; and we argue that there should be more research contribution in the particular area of testing and maintenance.

Keywords: Systematic Mapping Study, Mobile Apps, Cross-Platform Development, Literature Review

Introduction

Nowadays, smartphones are becoming widely used in different life aspects such as business, education and entertainment to mention just a few. Moreover, mobile apps are being integrated into critical sectors such as m-banking, m-payments and health (Nagappan and Shihab, 2016). People use various types of smartphones, which means different platforms and wide range of devices specifications. There are at least four important platforms (iPhone android, BlackBerry and Windows Phone) and one problem for developers is to provide your applications for all these devices and platforms. Further, the area of mobile app software engineering emerged to be one of the most rapidly growing areas and mobile devices are quickly developing in both hardware and software aspects.

There are three different approaches to build mobile apps: native, mobile web (m-site) and cross-platform. Native mobile app development approach is a methodology through which building apps is done using the programming language for a specific platform. Additionally, native mobile apps operate only on the platform that they are built for, which means that developing native applications for different platforms needs to hire different teams, each for separate technology. This approach is not very attractive due to the cost, the number of resources and

time associated with the development activities for each platform. The developers must create and maintain the same application for each platform. One of the solutions is building responsive mobile web apps (m-sites) which is another approach for mobile app development. M-site apps are built using web technologies, mainly HTML, CSS and JavaScript and can be accessed through web browser. Further, m-sites are highly connected and they have limited access to mobile device hardware and sensors, which reduces the flexibility of building mobile app. An ideal solution to this problem is to use the approaches of multiplatform development, which allows the developers to implement their applications with a variety of features. The cross-platform development of applications includes different solutions such as hybrid, interpreted, cross-complied and other approaches. This approach provides solutions to facilitate building an app once then deploy it into a spectrum of platforms. Since the resulted app can be run over different platforms, it reduces the time and resources cost for development and increases the productivity.

Thus, the developers, who want to join the world of mobile apps development will face questions about differences between the different approaches, they will seek to learn more about the approaches from the literature. The main objective from this study is to answer such questions through systematic review of the literature.

As the best of our knowledge, there is no comprehensive systematic mapping study in the area of cross-platform mobile apps development. This provided a motivation for conducting a mapping study in this field. Our mapping study collected and described all the relevant studies in order to highlight the research gaps in the field. Initial search results produced 295, 816 results. However, after applying multi-step inclusion/exclusion criteria, a total of thirty (30) studies were finally included in this mapping (Appendix A). We present the studies based on four-part classification scheme (i) structure of the topic; (ii) contribution facets; (iii) applied techniques; and (iv) research facets. We argue that evaluation studies should be based on more complex and real-world apps; there is a lack of studies focusing on testing solutions/challenges for cross-platform app development; and more research should be directed on the maintenance aspect of cross-platform development.

The remainder of this paper is structured as follows: We introduced the motivation and the related work in Section 2. In Section 3 we described the methodology of this mapping study. The results is presented in Section 4 followed by Section 5 where we discussed our findings. Finally, we concluded our work in Section 6.

Motivation and Related Work

To our best of knowledge, the study by (Amatya and Kurti, 2013) is the only study that provides a survey about the trends in cross-platform development. In their study, the authors mainly focused on the raised development issues in cross-platform field. Further, they did not include the studies that investigated development tools or approaches. Moreover, their study did not critically review included studies; instead, they merely presented identified problems with the proposed solutions in each study. In contrast, our study includes the most recent research in the area of cross-platform development. Additionally, we covered all the relevant studies with different areas of interests and we provide a brief summary for each study.

In another study conducted by (Latif *et al.*, 2017), the authors gave an overview of the recently applied development tools and approaches. They provided a recap for the cross-platform approaches and their platforms. The described approaches are: web, hybrid, interpreted, cross-compiled and model-driven approaches. They compared the approaches based on their use and most popular platforms. They concluded that the hybrid approach is good for developing low complex features. Further, they argue that cross-compiled platforms are good for enterprises development work, since it requires developing once then can be deployed everywhere as native app.

Another study conducted by (Ribeiro *et al.*, 2012) provided a global view of the cross-platform development technologies and highlighted their pros and cons. In their study, six different tools have been analyzed and compared against each other. The analyzed tools are: Rhodes, PhoneGap, DragonRAD, Titanium, mobile and mdl. The study identified the strength and weakness points for each tool, it also discovered that the majority of the tools do not apply model driven engineering, only two of them applied it.

Alamri and Mustafa (2014) conducted a study that identified the challenges in cross-platform development. They summarized that the major issues are (1) poor performance; (2) low user interface quality; (3) limited access to hardware features. A study of (El-Kassas *et al.*, 2014) surveyed the existing cross-platform tools to provide an overview for the most recent used tools. Their study can be used as a reference for the developers in cross-platform area. It also provided the open issues in this area which are (1) find a solution that is compatible for different mobile platforms; (2) support the native programming language and consider the APIs differences; (3) support the native user interface; (4) support code reuse.

In the area of mobile development, (Samer *et al.*, 2016) conducted a systematic mapping study in mobile apps testing techniques, which we inspired the design of our mapping study. This study categorized 79 empirical studies based on four classification scheme (structure of the topic, contribution facets, objects involved in the study and research facets), which the authors inspired these categories from other studies that provided a guideline for systematic mapping studies. Further, it provided an in-depth overview for the mapped studies using a well-defined methodology. They identified several research gaps in the field of mobile apps testing and highlighted major issues in the testing process: a need for formulating the testing requirements early; lack of studies conducted in real-world development environments; further research should be conducted in the area of testing life cycle models and mobile services.

Another systematic mapping study is conducted in mobile development field by (Tramontana *et al.*, 2018). They presented an overview for the tools and techniques that support automating the process of mobile apps functional testing through classification scheme, which included several attributes and sub-attributes such as support to test automation, testing inputs, evaluation and others. They included 131 papers that were classified based on the supported testing activities; the testing techniques characteristics they present; and the adapted evaluation methodology. This study concluded with some research gaps and trends, such as, lack of industry contribution in the research.

There is no rigorous systematic mapping study in the area of cross-platform development. We found only one literature survey (Amatya and Kurti, 2013) which is conducted in a very early stage of cross-platform development. We found relevant studies that provided an overview for the existing development tools (e.g., (Latif *et al.*, 2017), (Ribeiro *et al.*, 2012) and others). Further, we found some studies that reported the challenges in this field such as (Alamri and Mustafa, 2014); (El-Kassas *et al.*, 2014). This motivated us to apply a rigorous systematic mapping study in order to provide a wide overview for the existing and most recent research in this emergent field.

Method

This study applied the systematic mapping method by following the guidelines provided in (Petersen *et al.*, 2008). Further, the design of our study is inspired by other systematic mapping study by (Zein *et al.*, 2016). In general, systematic mapping studies objective is to provide an overview for the existing research in a specific field. For this, our systematic review objective is to identify studies that conducted in the field of cross-platform development techniques for mobile applications and obtain a summary of the technologies and methodologies used in this type of development.

As described by (Petersen *et al.*, 2008), a systematic mapping study has five phases. The first phase is specifying the research questions. The second phase is the search process where the researcher follows a pre-defined search strategy to search for and select studies. The third phase is about skimming the studies, then extracting keywords from the abstracts during the fourth phase. The last phase is extracting the data from the studies and mapping them into the defined scheme. Figure 1 shows the steps and phases of the systematic mapping review.

Research Questions

This study built a classification scheme based on the included studies in the field of cross-platform mobile app development. Moreover, we presented an overview of the existing research and tried to find the research gaps and challenges in order to guide the future research in this field. We specified four research questions to achieve the goal of this mapping study. The research questions have the objective of finding studies to understand and summarize the use of cross-platform development approaches to mobile applications, to this end, the following been established: The primary question is to identify the contribution for each study included in this mapping study. Then, we stated three sub-questions to present the main

achievement; main challenges that have been investigated; and present the main research techniques they applied:

- **Primary RQ:** What are the empirical studies that have been done in the area of mobile app cross-platform development?
- **Sub-RQ1:** What accomplishments do these studies present and what contribution facets do they provide?
- **Sub-RQ2:** What are the main challenges addressed by these studies?
- **Sub-RQ3:** What are the main research techniques applied by these studies?

Source of Evidence

The present study was conducted at Birzeit University in Palestine. We have defined as sources of research information the following online libraries provided by the university, namely: IEEEExplore, ACM Digital Library and Google Scholar.

Search Strategy

Included studies in this review are from both quantitative and qualitative approaches, they are all related to cross-platform mobile app development. Aiming to establish the search strategy, considering the research questions, we adopted the strategy proposed in the study of Kitchenham and Charters (2007):

- Use the Boolean operators OR and AND:
 - Use AND to limit the search by major terms
 - Use OR to broaden the search
- Search for alternative spellings

In our search process, it took several attempts to finalize the good search string, which has been evaluated according to the number of retrieved studies and their relevance. Our challenge was handling the fact the mobile word is a general term that could refers to many things rather than the smartphone, cross-platform term is used in different contexts not only in mobile development and the words application and app are not equals in the search, each one could returns different studies.

The terms included in the search string were inspired from the proposed research questions. The terms “cross-platform mobile applications” formed the main search terms. We included the synonym for the “cross-platform” term by using “hybrid” term and “app” as the alternative for applications.

Based on the adopted search strategy and the defined search keywords, we formulated the following search string: ((Cross-platform OR Hybrid) and (mobile) and (applications OR apps)), (Appendix B).

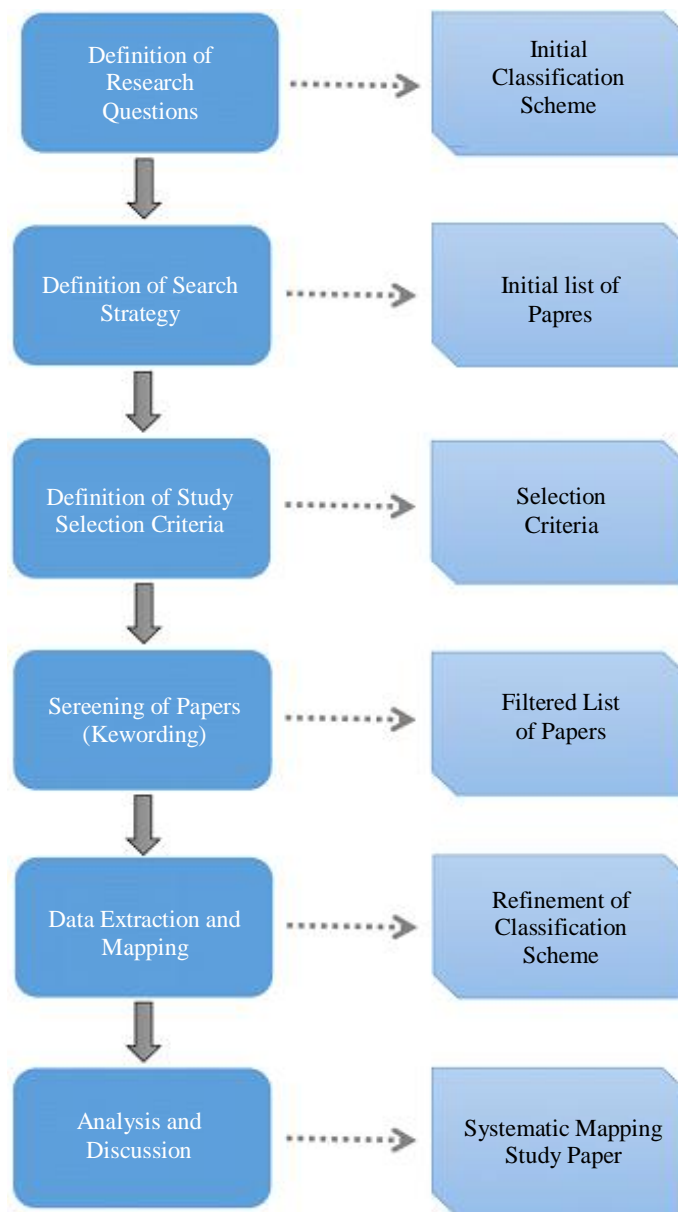


Fig. 1: Systematic mapping process

Additionally, we followed the search approaches mentioned in (Felderer *et al.*, 2018) which are:

- Backward Snowballing: Get the studies that are cited by the other studies included in the references
- Forward Snowballing: Get the studies the are cited in the existing studies

Study Selection and Criteria Process

In a systematic review, the definition of inclusion and exclusion criteria is important for include primary

studies that are relevant and will achieve the research objectives and exclude those irrelevant. The search process was done using the advanced search mechanism provided by the online libraries. We restricted our search to cover only the computer science field since hybrid term is widely used in other engineering fields.

The research studies selection phases were iterative and incremental, each study passed through three different stages in order to filter the studies which will be included in the mapping study. The earlier phase was applying a search string to get the relevant studies. Next in the first phase, we excluded

the papers that are not relevant to cross-platform mobile development, the decision is based on their titles and abstracts. In the second phase we applied selection criteria. We excluded the old papers that have been published before 2013. Further, we excluded short studies that are less than five papers. Additionally, we excluded the non-empirical studies by reading the introduction, methodology and conclusion. The final steps were completed by reading the remaining papers (Fig. 2).

Keywording of Abstracts (Classification Scheme)

The keywording process is performed based on the guidelines provided by (Petersen *et al.*, 2008), this process is done through two steps. The first step is to read the abstracts of the selected papers in order to find the main keywords, which indicate the studies contribution. If the abstracts do not have the good quality that allows keywords extracting, then researcher can read the introduction and conclusion sections. The goal of this process is to build the classification scheme, which represents the included studies.

The second step is applying a thematic analysis approach in which the chosen keywords from different studies are combined together to build a high-level understanding of the selected papers contribution. This step led to identify a set of categories for the classification scheme.

The keywords and concepts resulting from the first phase reflected the contribution for the papers. Example of concepts we got are compare cross-platform approaches; analyze cross-platform approach; introduce new approach; impact of using cross-platform; and cross-platform challenges and opportunities. During the second phase, the resulted concepts were combined together to identify the contribution area for each paper. The results were the main categories in the classification scheme, which are “comparative analysis”, “framework analysis” and “new approach”.

Data Extraction and Mapping of Studies

The aim of this phase is to analyze the selected studies to extract the data required to answer the

proposed research questions. The primary studies were sorted according to the defined classification scheme. Data was organized in different tables; the relevant frequencies for each category were calculated. The extracted data addressed the accomplishments and the contribution facets, the applied techniques and the main challenges.

The main challenges were extracted from the primary studies and summarized according to the identified categories. In addition, we extracted the research methods applied in the selected studies. The defined classification scheme gives a coherent overview of the included studies.

Results

Search Results

We did several attempts to make sure that we have the most reliable search string. As mentioned earlier, our search strings are based on “mobile”, “applications”, “apps” and “hybrid” terms. Finally and after different attempts with different pilot search strings, we agreed on the final one that could be considered an appropriate since it has all the necessary terms and almost all the retrieved research studies are resulted from this search string, which is “cross-platform/hybrid mobile application/app”.

At the beginning, our search results retrieves 295, 816 studies from all sources as mentioned in Section 3.2. After that, we applied the three filtration steps to decide what studies will be included in the mapping study. Table 1 shows the online libraries we searched in, the filtration steps we applied and their results. At the end, 30 studies have been included in the mapping study, 13 studies retrieved from ACM library which forms the highest percentage (43%) in the total results. And, 7 studies (24%) came from the IEEE library, while 10 studies (33%) resulted from google scholar.

Figure 3 shows the distribution of the selected studies according to the publication year. The earliest studies are those published in 2013 and most of the studies are published in 2016.

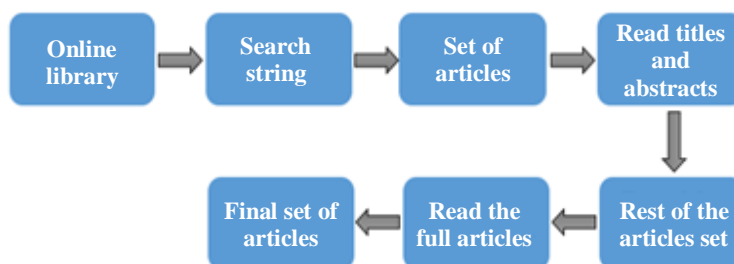


Fig. 2: Studies selection process

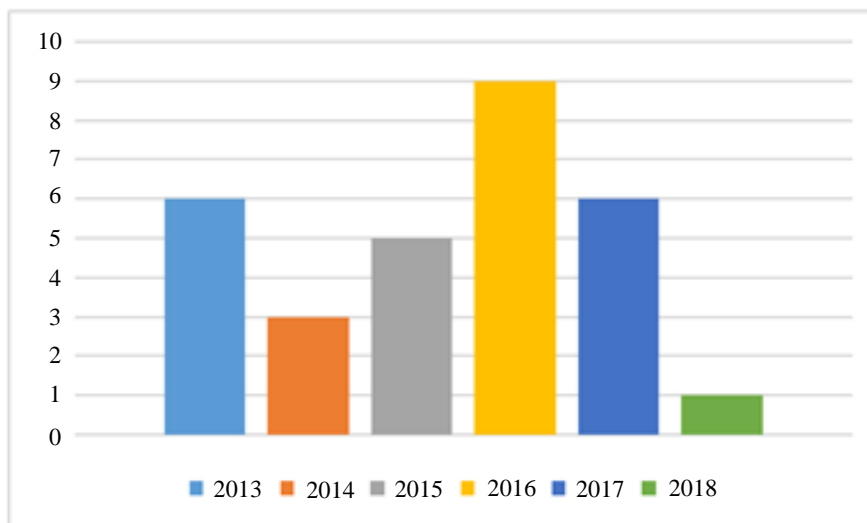


Fig. 3: Studies distribution based on the publication year

Table 1: Studies in each filtration step

Online Database	Search Results	Phase 1	Phase 2
ACM Digital Library	200,080	25	13
IEEEExplore	2736	23	7
Google Scholar	383,000	16	10
Total	295,816	64	30

Classification Scheme

We defined a classification scheme of four categories: (i) structure of the topic; (ii) contribution facets; (iii) applied techniques; and (iv) research facets. The first category (structure of the topic) is defined based on the thematic analysis, it is classified into three sub-categories: (i)“comparative analysis”, (ii)“framework analysis” and (iii)“new approach”. These sub-categories were identified during the keywording phase according to the main topic investigated in each included study.

The second category that forms the answer for sub-RQ1 is the contribution facets, which are inspired from the study by (Shahrokni and Feldt, 2013). We concluded with six contribution facets: metrics, tool, framework, review, evaluation and method. The study that presents a guideline in a specific area to be followed belongs to metrics facet. Those, which provide specific software tool for development or testing, are grouped under tool facet. The studies that provide an overview for a set of existing studies in the field are classified under review facet. Evaluation facet, which contains the highest number of studies, all, provided a comparison for tools, approaches or frameworks. The studies that explain specific goal or research questions belong to method facet.

The third category is concerned with applied techniques, answers the sub-RQ3. The empirical studies were classified based on their research

method, more specifically: experiment, survey, case study, interviews and mixed methods. Most of the included studies applied an experiment and two of them supported their experiment with survey. Further, only one study conducted interviews.

The fourth category is inspired from (Petersen *et al.*, 2008) which classified the studies based on the research facets. It contains three research facets, validation research, evaluation research and solution research:

- **Solution research** are those propose a solution and provide a discussion about how it’s doable without validate it
- **Validation research** are the studies that provide an investigation for the new proposed solution that are not implemented in practice yet
- **Evaluation research** facet contains the studies that provide an investigation for some implemented technique or for known issue

Answering the research questions

Primary RQ: *What are the empirical studies that have been done in the area of mobile app cross-platform development?*

The studies presented in this mapping study have been categorized based on the discussed classification scheme in Section 4.2. The basic category to cover and highlight the main research topic includes comparative analysis, framework analysis and new approach.

Table 2: Studies in each filtration step

Category	Studies (S)	Total#studies
Comparative Analysis	S2, S4, S5, S6, S7, S10, S11, S12, S14, S15, S16, S17, S21, S25	14
Framework Analysis	S1, S3, S9, S13, S18, S23, S26	7
New Approach	S8, S19, S20, S22, S24, S27, S28, S29, S30	9

This study covered thirty (30) research studies, fourteen (14) of them are under comparative analysis category, seven (7) studies grouped under framework analysis and nine (9) studies came under new approach category. Table 2 shows the mentioned categories and the studies belong to each one, it is clear that the highest number of studies are those under comparative analysis.

Comparative Analysis

Past comparative studies have been done on different ways, the applied research methods differs for the studies. Some studies executed the comparison through an experiment, others with case study, few of them using survey and some of them decided to support the main method with another one so they done the evaluation using mix of methods.

The contribution for each study varies from one to another, some studies compared the cross-platform tools against each other and against native development tool, and other studies compared the applications that have been developed in a hybrid way against those developed using the native technologies. This section presents 14 studies that conducted a comparison analysis S2, S4, S5, S6, S7, S10, S11, S12, S14, S15, S16, S17, S21 and S25.

Taneja *et al.* (2016) [S2] analyzed different development approaches that can be used to achieve the platform independence. Three approaches have been discussed and compared against each other in order to test their performance and decide the approach that can solve the platform dependence. Virtual Machine based, Distributed Computation based and Hardware based approaches have been evaluated according to two performance parameters, which are throughput and response time. They executed the experiment on four different applications that vary in their complexity and length. The best throughput and response time are achieved from the hardware-based approach, and the worst throughput and delay record was for the distributed computation based approach, while the virtual machine based performance was in between the previous approaches.

The survey applied by (Latif *et al.*, 2016) [S4] presented a comparison of cross-platform development approaches. It analyzed five approaches, web, hybrid, interpreted, cross-compiled and model driven approach. It has also discussed the desirable features for the eligible development tool, which consider the scalability and maintainability of the application, the access

capability to device features, the resources consumption, the security level and the development environment. They concluded that each approach has its advantages and challenges, but the cross-platform development can be considered as the best approach when the time and cost are limited.

Heitkötter *et al.* (2013) [S5] evaluated the development approaches of cross-platform apps, which are the Web apps and the apps developed by PhoneGap and Titanium. The evaluation has been done through interviews with experienced developers and based on some criteria that was divided into two sections, one is related to the infrastructure perspective such as the license, cost, and supported platforms and more, the other is about the development perspective like ease of development, maintainability, scalability and others. After that, they compared those apps with native apps. This evaluation study found that developing the apps natively is not necessary since the cross-platform approaches are mature enough to cover the needs of building mobile apps.

The study of (Xanthopoulos *et al.*, 2013) [S6] investigated the cross-platform application types, which are web, hybrid, interpreted and generated apps. Then, a comparative analysis is done based on the historical review of each app type and according to set of criteria. They concluded that there is no perfect development approach. Further, the approach selection decision should be taken with considering the needed requirements and expectations of the app. For example, web apps are a good start, which can be used in other approaches. The hybrid and interpreted apps are must when the publication on the store is required. A case study is applied by designing an interpreted application of RSS reader, the application is created using Titanium tool. The case study supports that the interpreted approach is promising. They built the application using JavaScript without any experience in the environment platform.

The comparison analysis done in (Wilcox *et al.*, 2016) [S7] evaluated an application that has been built using both native and hybrid technologies, it is PropertyCross app. This research applied a quantitative analysis approach to measure the performance of the cross-platform development tools. The measurements parameters are the response time, CPU usage, memory usage and disk usage, they have been calculated on five different devices with the three main platforms android, iOS and Windows Phone. This study discussed several findings that can help

developers in selecting the tool when developing a hybrid mobile application. The cross-platform tools have the most CPU and memory usage and they recorded the slowest app launch time, but they respond as fast as native apps during pages navigation.

Delia *et al.* (2015) [S10] presented a case study that discussed the most important features of four different cross-platform development approaches. The analyzed cross-platform approaches are mobile web, hybrid, interpreted and cross-compiled. The presented case study is a web application designed only for desktops named WebUNLP application, which is a virtual environment for teaching and learning, it contains many features to enable sharing materials and facilitate the communication between the educators and students and other beneficial features. As part of the case study, one feature has been chosen to be adopted into mobile platform that is the electronic notice board. The feature is developed with the all cross-platform approaches mentioned above. In case of mobile web application, it was simple to develop and deliver, but the notifications are not received when there is new post on the board. Regarding the hybrid application, it was developed by PhoneGap, which provides the simplicity of web approach and use all device features. Although its performance was worse than the native app but it was much better than the mobile web app. Interpreted version was developed using Titanium, it was helpful in achieving a high level of performance by generating the code for the interface. However, it does not afford a tool that enables interface design visualization. Finally, the cross-compiled application was developed using Xamarin/Visual Studio and Delphi XE6. Fully native versions were obtained in both technologies.

A survey of different cross-platform tools is conducted in (Dalmasso *et al.*, 2013) [S11], it covered the PhoneGap, Titanium and Sencha Touch tools. The conducted survey used example of applications developed by the mentioned tools and then had been evaluated in terms of CPU, memory usage and power consumption. The results of this survey showed that the PhoneGap utilized less CPU and memory and consumed less power than Titanium. Moreover, PhoneGap and Sencha Touch are working very well together when the memory availability is not an issue; they also provide a good UI design.

A comparison analysis for hybrid apps and their tools is conducted in (Ali and Mesbah, 2016) [S12]. In their study, 15, 512 hybrid apps were collected from the stores; they identified what Cross-Platform Tools (CPT) they used and how users rate them. Then, the rating of hybrid apps is compared against the native apps rating from the same category. At the end, they compared the hybrid app rating for Android against iOS. The success of the CPT is measured according to the number of reviewers, rating and number of downloads for the app in the store that have been analyzed through four

research questions. Although PhoneGap tool is the most popular CPT in the app stores with 86% but the apps developed by AdobeAir CPT have more downloads and reviews number. Despite that, higher downloads and reviews number doesn't indicate user satisfaction from the apps, PhoneGap CPT got better user reviews.

An experiment is done in (Vilcek *et al.*, 2017) [S14] by building a simple mobile app that does simple calculation which is adding two numbers. This app was developed using six different IDEs android Studio, Xcode, Visual Studio, Ionic, PhoneGap and NativeScript. In order to explore the advantages and disadvantages for these tools, they were compared according to the some criteria such as supported computer OS, supported mobile OS, programming languages and others. Android Studio has an advantage of ability to develop applications on all computer operating systems. Based on the analyzed criteria, hybrid development tools were considered as reliable tools, PhoneGap is one of the hybrid tools that has good advantages due to number of supported mobile platforms.

Majchrzak and Grønli (2017) [S15] assessed three cross-platform development tools through design-oriented experimental analysis combined with survey to support the findings. The study analyzed the ReactNative, Ionic and Fuse frameworks, they have been assessed according to survey result of 100 responses and then, prototype app is designed based on the survey results. This practice-assessment analysis focused on the user experience through developing a real-world use case, it is also considered the remote fetching data, navigation and developer experience as a comparison factors. The analyzed cross-platform tools have common problems related to compatibility, performance, testability and others. ReactNative developers had to handle many architectural decisions while the Ionic framework made it easier by providing the component library.

The experiment conducted by (Angulo and Ferre, 2014) [S16] aimed to evaluate the impact of creating app using cross-platform tool on the UX. The study done by having two independent development teams, one developed a Titanium app version and the other built two native versions. The evaluation results showed that there is no extreme differences in the UX between the developed versions, especially in the Android platform.

Ali *et al.* (2017) [S17] implemented a comparative study using mixed-methods, quantitatively and qualitatively. They used open source tools to mine apps from both Google Play Store and Apple Store, 80K hybrid apps were collected from the stores. These apps were compared according to user reviews, versions, prices and other factors. The analysis was done in order to get a better understanding about the challenges that developers face in developing these kinds of apps and what makes user rates the same app differently over different platforms. The results were concluded based on the qualitative feedbacks from the users and the quantitative

research questions. They found that the most developed apps are those built for specific platform, additionally, although most Android apps are free but the paid apps have higher prices than the iOS paid apps. More than 80% of the top-rated apps are the hybrid apps while the Android versions for the pair-apps have higher user rating.

After conducting an analysis study on hybrid apps based on developers' perception, (Malavolta *et al.*, 2015) [S21] conducted another study to analyze the hybrid apps according to end users perception. They mined 11, 917 apps from the Google Play Store with 3,041, 315 user reviews. Those apps were analyzed by studying the users' reviews for the hybrid apps compared with the reviews for the native apps. The analysis done through four qualitative research questions, all of them focused on the users' perception of the hybrid apps with respect to their perception of the native ones. The study found that the hybrid development frameworks have a good perception when they used for mobile apps of data-intensive, but they perceived poorly when building apps that require an access to system features. The end user value for both native and hybrid is the same. Additionally, hybrid apps in some categories get lower perception than the native apps.

The study by (Ciman and Gaggi, 2017) [S25] conducted an experimental analysis to compare the existing cross-platform frameworks according to energy consumption. They tried to prove that the adoption of specific cross-platform development framework can affect the energy consumption. The experiment used hardware and software setups and the used applications are for both Android and iOS. Each application is executed for 2 minutes and it is repeated for three times. The results confirmed that developing using cross-platform approach is absolutely reveals an increase in the energy consumption.

Additionally, the studies under comparative analysis category are classified into sub-categories according to the object involved in each study. The comparison in these studies is done on different objects, these research objects are (I) approaches, which identify the differences between various development approaches, such as hybrid vs. native; (II) frameworks which investigated the developed apps based on the used framework, e.g., JavaScript frameworks vs. code translation frameworks; (III) tools, the evaluation is done based on the development tool used to build the applications such as PhoneGap vs. Titanium; (IV) application attributes, the apps are evaluated according to their attributes, e.g., user reviews and number of downloads. Table 3 shows each sub-category and its relevant studies.

Table 3: Comparative analysis sub-categories

Research Object	Study
Approaches	S2, S4, S5, S6, S10, S16, S25
Frameworks	S7, S15
Tools	S11, S12, S14
Applications Attributes	S17, S21

Framework Analysis

Part of the applied studies analyzed the cross-platform development in different ways, some conducted a survey or a case study that focused on the development process and others analyzed the developed hybrid apps. Mainly, the purpose for those kinds of studies was to identify the challenges and opportunities for hybrid development approach. In this section, seven relevant studies will be discussed S1, S3, S9, S13, S18, S23 and S26.

The study by (Martinez and Lecomte, 2017) [S1] analyzed the process of building cross-platform mobile apps, studied the maintenance process for the cross-platform apps in terms of frequent discovered bugs and how they are fixed, finally, the study tried to define a tool to automatically fix the bugs. The analysis study is done through an experiment that built mobile apps using Xamarine and React-Native frameworks.

Amatya and Kurti (2013) [S3] conducted a literature survey study in order to assess the trend for cross-platform development in the last few years. They used the keywords of "mobile" and "cross-platform" to mine the relevant research papers from ACM and IEEE libraries, then, they applied some inclusion criteria and left with 17 articles. Finally, they showed the research results by describing the cross-platform development problems that each paper addressed and the proposed solutions with the used technologies to handle the problem. The results showed that despite that, the cross-platform development tools are not highly powerful yet, but they are promising. The impact of using the multi-platform development tool on the users' reviews is studied by (Mercado *et al.*, 2016) [S9]. This study attempted to find out if there was any relationship between the specific development tool and the user's perception that reflects the quality of the application. They mined 50 apps developed using three different hybrid approaches and analyzed 787, 228 user reviews in terms of security, performance, reliability and usability. They found that is selecting the development approach is absolutely affects the users' perception.

The study by (Dunka *et al.*, 2017) [S13] investigated developing hybrid mobile apps using Ionic framework. They analyzed the Native, Web and Hybrid mobile development approaches before explaining how the apps can be developed using Ionic framework technologies. The technical details presented in this study can be used as a reference for building enterprise level application. The analysis study applied in (Vitols *et al.*, 2014) [S18] aimed to address the issues for building hybrid apps using PhoneGap and tried to provide solutions for these issues. The analysis has been done by using the PhoneGap tool to implement features of Baltic Insurance House mobile application, the development covered different functionalities. The result confirmed that

several points should be considered when choosing PhoneGap as the development tool.

The study by (El-Kassas *et al.*, 2014) [S23] aimed to provide an overview for the available cross-platform approaches. They described each approach in details, which are, Cross-Compilation, Virtual Machine, Model-Driven Development, Web-Based, Component-Based, Cloud-Based and Merged approach. The study analyzed the pros and cons for them and proposed a categorization for many approaches. (Bjørn-Hansen *et al.*, 2017) [S26] surveyed the literature and introduced a technical baseline to be followed in the future research in the cross-platform development field. They suggested some approaches and frameworks to be included in the research such as Model-Driven and Component-Based approaches. The suggestions are formed based on the traversed literature and considered the most common discussed in the previous research, hence, researchers can avoid the deprecation.

New Approach

Some studies proposed new approaches to develop hybrid mobile apps; they presented the technical and architectural details for these approaches. The studies presented in this section are S8, S19, S20, S22, S24, S27, S28, S29 and S30.

Heitkötter *et al.* (2013) [S8] introduced a model-driven approach md2 for developing cross-platform apps. This approach helps the developers by generating the native code automatically after only specifying the high-level design for the app. They described the general concepts of the model then demonstrated the technical details of how the model can be implemented. Md2 has been discussed according to real-world project and proved that it is helpful for app development approaches.

The approach proposed by (El-Kassas *et al.*, 2014) [S19] is a mixed solution between some different cross-platform development approaches. They combined the advantages from those mixed approaches and tried to reduce their drawbacks. The proposed architecture with name of Integrated Cross-Platform Mobile Development (ICPMD) supported the source code availability (e.g., iOS) and of course the ability to run it on different platforms.

An evaluation framework analysis is presented in (Ahti *et al.*, 2016) [S20], it has been validated through an experiment in order to propose a decision-making helper. The framework validation aimed also to analyze a cross-platform tool, PhoneGap, by comparing it with native development tool for Android and Windows Phone platforms. They developed an application with two versions for Android and Windows Phone using native development tool and one hybrid version using PhoneGap. The proposed evaluation framework is used to assess these applications by comparing the native version against the hybrid version. This experiment

reveals that developing the hybrid version was easier than the native ones, but it showed an average quality in appearance and usability.

Brucker and Herzberg (2016) [S22] presented an approach of static code analysis that is a technique to detect the data-flows in hybrid mobile applications by generating calls graphs. This approach has been evaluated by applying it on apps mined from Google Play, these apps were developed using Cordova framework and ranked as top Cordova apps in the store. The results showed the generated calls graphs for cross-platform language are highly precise. The quality depends in the used language, which is JavaScript in this case.

The study by (El-Kassas *et al.*, 2016) [S24] extends [S19] by introducing a new code conversion approach, this approach proposed a way to find a matching for set of code patterns inside the input source code in order to produce the required source code for a specific platform. The study evaluated the efficiency of the generated apps from the new version of ICPMD compared with the apps developed using other cross-platform tools. The results showed a remarkable enhancement in the speed, memory usage and the app size.

A Native-2-Native approach is proposed by (Byalik *et al.*, 2015) [S27], it is a method to transform the source code of a specific platform to be compatible with another platform. This technique is applicable only for transforming the Android source code (Java) into iOS version (Swift). The code transforming done via plugin added to the Eclipse IDE, it is evaluated by applying the method on several Android/Java APIs. The results indicate that Native-2-Native is a promising tool to support cross-platform development.

(Chadha *et al.*, 2017) [S28] applied a study that extends [S27], it enhanced the proposed technique of (Native-2-Native) by made it able to transform the native source code from iOS/Swift into the Android version (Java). They improved the Eclipse IDE plugin to detect the code in Java or Swift that access the native resources and then create a query to get a resource in web-based programming that can access the equivalent native resource in Android or iOS platforms. The enhancement has been evaluated; it has also showed a valuable development tool.

The study done by (Tung *et al.*, 2018) [S29] proposed a library to facilitate the development of cross-platform apps with acoustic sensing. The solution library of name LibAcousticSensing (LibAS) has been evaluated in this study by developing three apps using it. These developed apps covered the main aspects of acoustic sensing such as sound fingerprinting, inter-device interaction and others. The evaluation findings showed that LibAS reduced the required lines code, hence, less effort and less development time.

Table 4: New approach sub-categories.

Framework type	Studies (S)
Development Framework	S8, S19, S24, S27, S28, S29
Evaluation Framework	S20, S22
Testing Framework	S30

The proposed framework by (Boushehrinejadmoradi *et al.*, 2015) [S30] is to test the development frameworks used to build cross-platform apps. The study built a prototype of the suggested tool named X-Checker and then tried it on the Xamarine framework. The tool caught 47 bugs in that framework. X-Checker tool is designed based on differential testing method, which has been proved that it is an effective testing method.

The proposed solutions introduced different types of new approaches, some of them proposed a development framework, others suggested an evaluation framework that can help in evaluating the developed apps using cross-platform frameworks and one study proposed a testing framework. A sub-categorization is showed in Table 4 according to the type of suggested framework.

Research Accomplishments and Contribution Facets

Sub-RQ1: *What accomplishments do these studies present and what contribution facets do they provide?*

The achievements for the presented research covered several aspects; several studies compared cross-platform development approaches and tools against each other. The comparison done according to the most important factors that developers should consider when developing a hybrid app and they affect the user perception of the app. The comparative studies results explained the advantages and disadvantages for the compared approaches or frameworks.

Part of the included studies analyzed specific development framework, they showed technical details about how apps can be developed using these frameworks. Moreover, Pros and Cons for each framework are explained. The comparative and analysis studies aimed to be used as a reference for the developers, they can benefit from the evaluation results that studies provide. Additionally, through the technical details and recommendations that have been included in the studies.

Some other studies proposed a solution in the field of cross-platform apps, most of the solutions have been validated through an experiment or other research method. The suggestions differ from each other, the majority of them proposed a development framework, two of them proposed an evaluation framework and only one suggested a testing framework.

Table 5: Research approach facets

Research approach	Studies (S)	#of studies
Evaluation Research	S1, S2, S3, S4, S5, S6, S7, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S21, S23, S25, S26	22
Validation Research	S20, S22, S24, S27, S28, S29, S30	7
Solution Research	S8	1

The presented achievements for the mapped studies are classified according to the research approaches, which is summarized in Table 5. The research approaches have been decided based on the conducted studies. Three research approaches are identified, evaluation research, validation research and solution research. Most of the studies were conducted using evaluation research approach (73%).

The contribution facets classification described in this section was inspired by other mapping study (Zein *et al.*, 2016). The facets were classified according to their contribution into the following groups: evaluation, framework, review, tool, method and metrics. The majority of the studies are classified under evaluation; they represented comparative studies to assess the approaches and the tools of the cross-platform development. Framework represents a detailed technique of wide purpose and covers several research questions. Review studies surveyed the literature to provide an overview for a specific area such as development approaches (Shahrokni and Feldt, 2013). A tool provides specific software for development or testing purposes. Methods explain certain goal and research question. Metrics, which we have only one metrics study in our mapping study, show a guideline to be followed in a specific area in order to achieve the goal. Figure 4 shows the distribution of contribution facets and Table 6 presents a summary of the contribution facets and their relevant studies.

The bubble chart below (Fig. 5) presents the number of papers grouped under the main categories, against the papers categorized based on the contribution facets.

Main Challenges

Sub-RQ2: *What are the main challenges addressed by these studies?*

We have presented three main categories for the covered studies in this mapping study, which are comparative analysis studies, framework analysis studies and new approach studies. The studies under each category addressed some relevant challenges; they are summarized in Table 7.

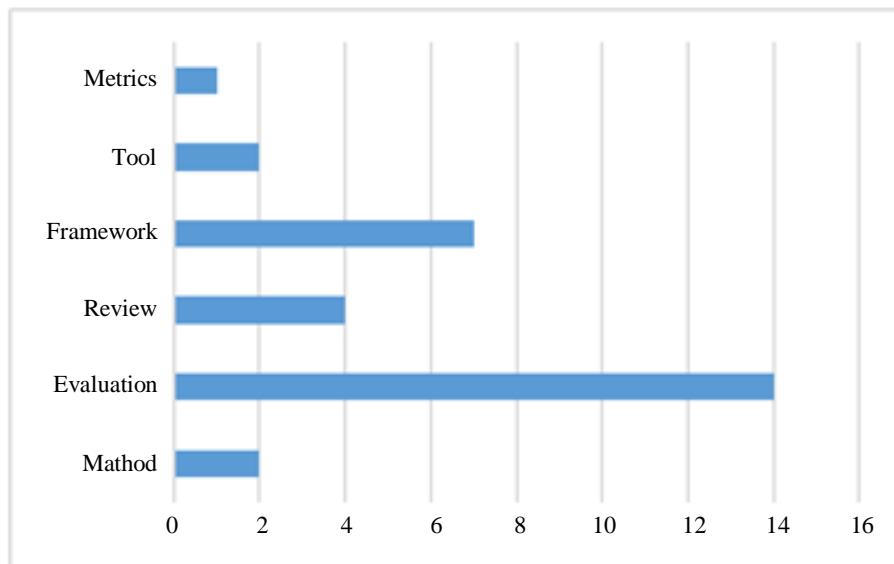


Fig. 4: Distribution of contribution facets

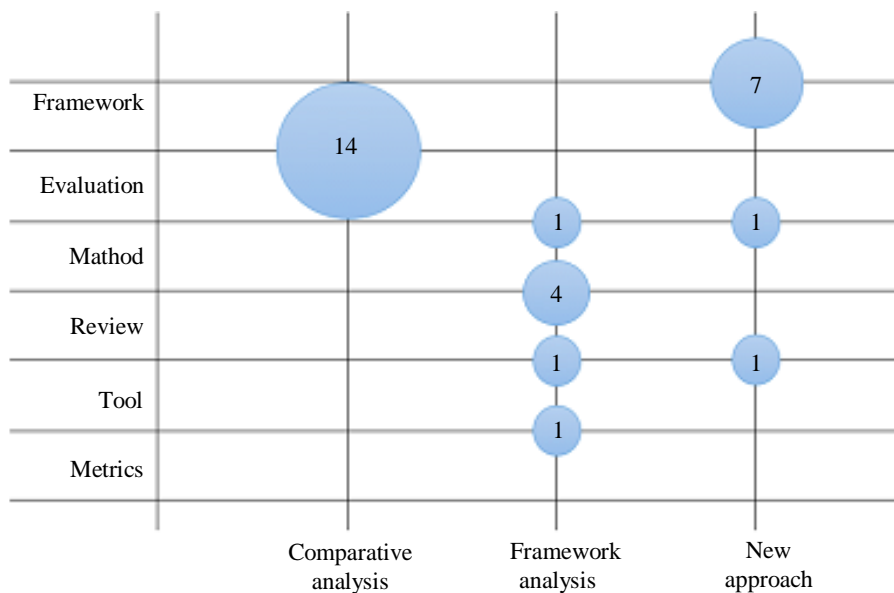


Fig. 5: Contribution facets against main categories

Table 6: Contribution facets and the mapped studies

Contribution facet type	Studies (S)	Description
Metrics	S26	Baseline to be followed in the future research in the cross-platform development field
Tool	S13	Ionic development tool to build hybrid apps for enterprise level
	S30	X-Checker tool to test the development frameworks used to build cross-platform apps
Framework	S8	Model-driven approach md2 for developing cross-platform apps by generating the native code automatically after only specifying the high-level design for the app
	S19	ICPMD is a development framework that supports the source code availability and the ability to run it on different platforms
	S20	Evaluation framework to assess the developed applications by comparing the native version against the hybrid version
	S24	Enhanced ICPMD, new code conversion approach to find a matching for set of code patterns inside the input source code

Table 6: Continue

	S27	Native-2-Native approach to transform the source code of a specific platform to be compatible with another platform, only from Android/Java into iOS/Swift
	S28	Enhanced Native-2-Native approach by made it able to transform the native source code from iOS/Swift into the Android version (Java)
	S29	LibAcousticSensing (LibAS) a library to facilitate the development of cross-platform apps with acoustic sensing
Review	S3	Conducted a literature survey study in order to assess the trend for cross-platform development in the last few years
	S9	The study attempted to find out if there an impact of using the multi-platform development tool on the users' reviews
	S18	Analysis study aimed to address the issues for building hybrid apps using PhoneGap
	S23	Provides an overview for these cross-platform approaches, Cross-Compilation, Virtual Machine, Model-Driven Development, Web-Based, Component-Based, Cloud-Based and Merged approach
Evaluation	S2	Analyzed different development approaches that can be used to achieve the platform independence
	S4	Presented a comparison of cross-platform development approaches, web, hybrid, interpreted, cross-compiled and model driven approach. Discussed the desirable features for the eligible development tool
	S5	Evaluated the development approaches of cross-platform apps which are the Web apps and the apps developed by PhoneGap and Titanium
	S6	A comparative analysis that is done based on the historical review of some app types which are web, hybrid, interpreted and generated apps
	S7	Comparison study to measure the performance of the cross-platform development tools based on the response time, CPU usage, memory usage and disk usage
	S10	Analysis of cross-platform approaches which are mobile web, hybrid, interpreted and cross-compiled, it evaluated the most important features of those development approaches
	S11	The study evaluated PhoneGap, Titanium and Sencha Touch tools in terms of CPU, memory usage and power consumption
	S12	Compared hybrid apps that have been developed using different cross-platform tools
	S14	Comparative experiment by building a simple mobile app that was developed using six different IDEs (Native and Hybrid)
	S15	Assessed three cross-platform development tools which are ReactNative, Ionic and Fuse frameworks
	S16	Conducted an experiment aimed to evaluate the impact of creating app using cross-platform tool on the UX
	S17	Analysis done in order to get a better understanding about the challenges that developers face in developing hybrid apps
	S21	Comparative study to analyze the hybrid apps according to end users perception (user' reviews)
	S25	Experimental analysis to compare the existing cross-platform frameworks according to energy consumption
Method	S1	Analyzed the process of building cross-platform mobile apps then tried to define a tool to automatically fix the bugs
	S22	Presented an approach of static code analysis which is a technique to detect the data-flows in hybrid mobile applications by generating calls graphs

Table 7: Addressed main challenges

Category	Challenge(s) addressed
Comparative Analysis	<ul style="list-style-type: none"> • Achieve UI consistency • Target the user experience of native app level • Build a maintainable app
Framework Analysis	<ul style="list-style-type: none"> • Build an application with a satisfactory level of performance • Create a real tool based on the existing framework to be used by the developers
New Approach	<ul style="list-style-type: none"> • Automatically integrate the generated code • Support multi types of applications • The ability to transform the code from the native source with considering the UI translation

Main Techniques

Sub-RQ3: What are the main research techniques applied by these studies?

Among the 30 studies, nineteen (19) of them applied an experiment to come up with findings by either answering research questions or investigate hypothesis. Moreover, there are another two studies that combined the experiment with survey, which means that experimental research formed the majority of the presented studies (about 70%). In contrast, only one study conducted interviews in its research process. Survey is conducted in another two studies, while case study is applied in only two studies. There are 4 studies that have reviewed the literature in order to give an overview for the existing studies. Table 8 shows the summary of the studies and applied techniques.

Discussion

This systematic mapping study aims to create a classification scheme and to collect, interpret and analyze all the relevant studies in the area of cross-platform mobile apps development. We found that there is no comprehensive systematic mapping study in this area, so, conducting such a study will help in improving the knowledge and spot the light on the gaps in the field.

The mapped studies showed some research gaps that require more investigation. First, most of the studies that were categorized under new approach category (Table 2) validate the proposed solutions on applications that have been built specifically for the validation purpose. If these studies were based on evaluating the solutions using real and more complex applications, then the scientific value of these studies would be more convincing and comprehensive. Some of these studies that used real applications mined from the app stores. However, these studies involved apps that are simple and do not represent the real-world applications. It is highly recommended to validate the proposed solutions and frameworks on more complex applications.

Secondly, the majority of the studies categorized under comparative analysis category conducted a comparison process based on the results of using native vs. hybrid mobile apps. However, these studies did not provide a detailed information about how they

distinguish between the apps types. More specifically, these studies did not explain how they were able to identify whether an app, which is downloaded from an online store, to be native or hybrid. If these studies gave the exact steps about how the differentiating process was done, then these studies would be more trusted and replicable.

Further, we found that there is a lack of studies focusing on testing challenges and solutions for cross-platform mobile apps. We found little studies that mentioned this challenge briefly and one study (Boushehrinejadmoradi *et al.*, 2015) [S30] that proposed a testing framework solution for the developed apps. Testing hybrid apps is an area that requires more investigation since there are several points should be considered when testing hybrid app. For instance, writing automation test cases can be easy if the app is built using only web-based technologies and with little native code, one test script can run across all supported platforms. On the other hand, how the testing process will be conducted when the app is developed with different languages? One possible solution is that each supported platform should has its own tests scripts. Such an issue and other testing issues should be further investigated in the field of testing cross-platform apps.

Finally, little studies investigate the maintenance challenges and solutions for the cross-platform apps. Maintainability can largely affect the developers' choice when deciding the approach to develop a mobile app. Additionally, mobile apps are rapidly changing because they should be up to date to stay competitive, which is achieved by adding new features and improve the existing ones. Therefore, we believe that researchers should contribute better in this area to create a reference for the developers as several aspects can be covered and investigated. The literature should show the developers how the cross-platform app affects the maintenance cost in terms of effort and time. Moreover, the research can cover the required skills for maintaining hybrid apps, whether the developers need specialized skills or not. Lower cost, less maintenance duration and minimal learning curve in the maintenance process are sufficient factors to develop apps using cross-platform approach.

Table 8: Applied techniques

Research technique	Studies (S)
Experiment	S1, S2, S7, S8, S9, S12, S13, S14, S19, S20, S21, S22, S24, S25, S27, S28, S29, S30
Survey	S4, S11
Case study	S10, S18
Interviews	S5
Mixed	S15, S17
Literature Review	S4, S7, S24, S26

Experiment+Survey

Conclusion

This paper presents a review of the state of knowledge of empirical studies in the area of mobile cross-platform app development. After analyzing 295,816 studies and applying multi-step inclusion/exclusion criteria, thirty (30) studies were finally included and mapped to a classification schema.

The classification contained four categories (1) structure of the topic; (2) contribution facets; (3) applied techniques; and (4) research facets. The category of structure of the topic is defined based on the thematic analysis and is classified into three sub-categories: comparative analysis, framework analysis and new approach. The contribution facets group included six categories: metrics, tool, framework, review, evaluation and method. The third category is concerned with the techniques applied in the selected studies categorized the empirical studies based on the research methods they used, experiment, survey, case study, interviews and mixed methods. The category of research facets classified the studies based on the research facets. It contains three research facets, validation research, evaluation research and solution research.

Two main research gaps were identified as described in section 5. First, there is a lack of studies focusing on testing challenges and solutions for mobile cross-platform app development. Secondly, little studies focus on the maintenance challenges. Both challenges are only mentioned briefly in the studies without conducting an empirical research.

Acknowledgement

The authors would like to thank Birzeit University for their support in this research.

Author's Contributions

Both authors contributed equally in this research.

Ethics

This research is original and contains unpublished material. The authors confirm that there are no conflict of interest involved.

References

Ahti, V., S. Hyrynsalmi and O. Nevalainen, 2016. An evaluation framework for cross-platform mobile app development tools. Proceedings of the 17th International Conference on Computer Systems and Technologies (CST' 16).

- Alamri, H.S. and B.A. Mustafa, 2014. Software engineering challenges in multi platform mobile application development. *Adva. Sci. Lett.*, 10: 2115-2118. DOI: 10.1166/asl.2014.5671
- Ali, M. and A. Mesbah, 2016. Mining and characterizing hybrid apps. Proceedings of the International Workshop on App Market Analytics, (AMA' 16).
- Ali, M., M.E. Joorabchi and A. Mesbah, 2017. Same app, different app stores: A comparative study. Proceedings of the IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (SES' 17).
- Amatya, S. and A. Kurt, 2013. Cross-Platform Mobile Development: Challenges and Opportunities. In: *Advances in Intelligent Systems and Computing*, Trajkovik, V. and M. Anastas (Eds.), ISBN-10: 978-3-319-01466-1, pp: 219-229.
- Angulo, E. and X. Ferre, 2014. A case study on cross-platform development frameworks for mobile applications and UX. Proceedings of the XV International Conference on Human Computer Interaction, (HCI' 14).
- Biörn-Hansen, A. G. Tor-Morten and G. Gheorghita, 2017. Baseline requirements for comparative research on cross-platform mobile development.
- Boushehrinejadmoradi, N., V. Ganapathy, S. Nagarakatte and L. Iftode, 2015. Testing cross-platform mobile app development frameworks (T). Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering (ASE' 15).
- Brucker, A.D. and M. Herzberg, 2016. On the Static Analysis of Hybrid Mobile Apps. In: *Computer Science Engineering Secure Software and Systems*, Caballero, J., E. Bodden and E. Athanasopoulos (Eds.), Springer, Cham, ISBN-10: 978-3-319-30806-7, pp: 72-88.
- Byalik, A., S. Chadha and E. Tilevich, 2015. Native-2-native: Automated cross-platform code synthesis from web-based programming resources. *ACM SIGPLAN Notices*, 51: 99-108. DOI: 10.1145/2936314.2814210
- Chadha, S., A. Byalik, E. Tilevich and A. Rozovskaya, 2017. Facilitating the development of cross-platform software via automated code synthesis from web-based programming resources. *Comput. Syst. Structures*, 48: 3-19. DOI: 10.1016/j.cl.2016.08.005
- Ciman, M. and O. Gaggi, 2017. An empirical analysis of energy consumption of cross-platform frameworks for mobile development. *Pervasive Mobile Comput.*, 39: 214-230. DOI: 10.1016/j.pmcj.2016.10.004
- Dalmasso, I., S.K. Datta, C. Bonnet and N. Nikaiein, 2013. Survey, comparison and evaluation of cross platform mobile application development tools. Proceedings of the 9th International Wireless Communications and Mobile Computing Conference (MCC' 13).

- Delia, L., N. Galdamez, P. Thomas, L. Corbalan and P. Pesado, 2015. Multi-platform mobile application development analysis. Proceedings of the IEEE 9th International Conference on Research Challenges in Information Science, May 13-15, IEEE Xplore press, Athens, Greece. DOI: 978-1-4673-6630-4/15
- Dunka, B., E. Edim and O. Dantala, 2017. Hybrid mobile application based on ionic framework technologies. *Int. J. Recent Advances Multidisciplinary Res.*, 04: 3121-3130
- El-Kassas, W.S., B.A. Abdullah, A.H. Yousef and A. Wahba, 2014. ICPMD: Integrated cross-platform mobile development solution. Proceedings of the 9th International Conference on Computer Engineering and Systems (CES' 14).
- El-Kassas, W.S., B.A. Abdullah, A.H. Yousef and A.M. Wahba, 2016. Enhanced code conversion approach for the Integrated Cross-Platform Mobile Development (ICPMD). *IEEE Transact. Software Eng.*, 42: 1036-1053. DOI: 10.1109/tse.2016.2543223
- El-Kassas, W.S., B.A. Abdullah, A.H. Yousef and A.M. Wahba, 2017. Taxonomy of cross-platform mobile applications development approaches. *Ain Shams Eng. J.*, 8: 163-190. DOI:10.1016/j.asej.2015.08.004
- Felderer, M. and C.C. Jeffrey, 2018. Guidelines for systematic mapping studies in security engineering. *Empirical Research For Software Security: Foundations and Experience*.
- Heitkötter, H., S. Hanschke and T.A. Majchrzak, 2013. Evaluating cross-platform development approaches for mobile applications. In: *Business Information Processing, Web Information Systems and Technologies. Applications*. Cordeiro, J. and K.H. Krempels (Eds.), Springer, Berlin, Heidelberg, ISBN-10: 978-3-642-36608-6, pp: 120-138.
- Heitkötter, H., T.A. Majchrzak and H. Kuchen, 2013. Cross-platform model-driven development of mobile applications with md2. Proceedings of the 28th Annual ACM Symposium on Applied Computing, (SAC' 13).
- Kitchenham, B. and S. Charters, 2007. Guidelines for performing systematic literature reviews in software engineering. School of Computer Science and Mathematics, Keele University.
- Latif, M., Y. Lakhrissi, E.H. Nfaoui and N. Es-Sbai, 2016. Cross platform approach for mobile application development: A survey. Proceedings of the International Conference on Information Technology for Organizations Development, Mar. 30, IEEE Xplore Press, Fez, Morocco. DOI: 10.1109/it4od.2016.7479278
- Latif, M., Y. Lakhrissi, N. El Habib and N. Es-Sbai, 2017. Review of mobile cross platform and research orientations. Proceedings of the International Conference on Wireless Technologies, Embedded and Intelligent Systems, Apr. 19-20, IEEE Xplore Press, Fez, Morocco. DOI:10.1109/wits.2017.7934674
- Majchrzak, T. and T. Grönli, 2017. Comprehensive analysis of innovative cross-platform app development frameworks. Proceedings of the 50th Hawaii International Conference on System Sciences, (CSS' 2017).
- Malavolta, I., S. Ruberto, T. Soru and V. Terragni, 2015. End users perception of hybrid mobile apps in the google play store. Proceedings of the IEEE International Conference on Mobile Services, (CMS' 15).
- Martinez, M. and S. Lecomte, 2017. Towards the quality improvement of cross-platform mobile applications. Proceedings of the IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems, May 22-23, IEEE Xplore Press, Buenos Aires, Argentina. DOI:10.1109/mobilesoft.2017.30
- Mercado, I.T., N. Munaiah and A. Meneely, 2016. The impact of cross-platform development approaches for mobile applications from the users perspective. Proceedings of the International Workshop on App Market Analytics, (AMA' 16).
- Nagappan, M. and E. Shihab, 2016. Future trends in software engineering research for mobile apps. Proceedings of the IEEE 23rd International Conference on Software Analysis, Evolution and Reengineering, Mar, 14-18, IEEE Xplore Press, Suita, Japan. DOI: 10.1109/saner.2016.88
- Petersen, K., R. Feldt, S. Mujtaba and M. Mattsson, 2008. Systematic mapping studies in software engineering. Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering, Jun. 26-27, ACM, Swindon, UK, pp: 68-77.
- Ribeiro, A. A. Rodrigues and D. Silva, 2012. Survey on cross-platforms and languages for mobile apps. Proceedings of the Eighth International Conference on the Quality of Information and Communications Technology, Sept. 3-6, IEEE Xplore Press, Lisbon, Portugal, DOI:10.1109/quatic.2012.56
- Shahrokni, A. and R. Feldt, 2013. A systematic review of software robustness. *Informat. Software Technol.*, 55: 1-17.
- Taneja, K., H. Taneja and R.K. Bhullar, 2016. Cross-platform application development for smartphones: Approaches and implications. Proceedings of the 3rd International Conference on Computing for Sustainable Global Development, Mar. 16-18, IEEE Xplore Press, New Delhi, India, DOI:978-9-3805-4421-2
- Tramontana, P., D. Amalfitano, N. Amatucci and A.R. Fasolino, 2018. Automated functional testing of mobile applications: A systematic mapping study. *Software Quality J.*

- Tung, Y., D. Bui and K.G. Shin, 2018. Cross-platform support for rapid development of mobile acoustic sensing applications. Proceedings of the 16th Annual International Conference on Mobile Systems, Applications and Services - MobiSys 18. DOI: 10.1145/3210240.3210312
- Vilcek, T. and T. Jakopec, 2017. Comparative analysis of tools for development of native and hybrid mobile applications. Proceedings of the 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, (TEM' 17).
- VITOLS, G., 2014. Issues of hybrid mobile application development with phonegap: A case study of insurance mobile application. Proceedings of the International Baltic Conference, (IBC' 14).
- Wilcox, M., J. Vossaert and V. Naessens, 2016. Comparing performance parameters of mobile app development strategies. Proceedings of the International Workshop on Mobile Software Engineering and Systems-MOBILESoft (ESM' 16).
- Xanthopoulos, S. and S. Xinogalos, 2013. A comparative analysis of cross-platform development approaches for mobile applications. Proceedings of the 6th Balkan Conference in Informatics (BCI' 13).
- Zein, S., N. Salleh and G. John, 2016. A systematic mapping study of mobile application testing techniques. J. Syst. Software, 117: 334-356. DOI: 10.1016/j.jss.2016.03.065.

Appendix A

List of Included Studies

Study Code	Citation	Library	Concept
S1	Martinez and Lecomte, 2017). Towards the Quality Improvement of Cross-Platform Mobile Applications. 2017 IEEE/ACM 4 th International Conference on Mobile Software Engineering and Systems (MOBILESoft). DOI: 10.1109/mobilesoft.2017.30	IEEE/ACM	Analyze cross-platform approach
S2	Taneja <i>et al.</i> (2016) "Cross-Platform Application Development for Smartphones: Approaches and Implications." 2016, DOI: 978-9-3805-4421-2	IEEE	Compare cross-platform approaches
S3	Amatya and Kurti (2013) "Cross-Platform Mobile Development Challenges and Opportunities." Springer International Publishing Switzerland, 2013, DOI: 10.1007/978-3-319-01466-1_21	Google Scholar	Cross-platform challenges and opportunities
S4	Latif <i>et al.</i> (2016). Cross platform approach for mobile application development: A survey. 2016 International Conference on Information Technology for Organizations Development (IT4OD). DOI: 10.1109/it4od.2016.7479278	IEEE	Compare cross-platform approaches
S5	Heitkötter <i>et al.</i> (2013). Evaluating Cross-Platform Development Approaches for Mobile Applications. Lecture Notes in Business Information Processing Web Information Systems and Technologies, 120-138. DOI: 10.1007/978-3-642-36608-6_8	Google Scholar	Compare cross-platform approaches
S6	Xanthopoulos and Xinogalos (2013). A comparative analysis of cross-platform development approaches for mobile applications. Proceedings of the 6th Balkan Conference in Informatics on-BCI 13. DOI: 10.1145/2490257.2490292	ACM	Cross-platform challenges and opportunities
S7	Wilcox <i>et al.</i> (2016). Comparing performance parameters of mobile app development strategies. Proceedings of the International Workshop on Mobile Software Engineering and Systems-MOBILESoft 16. DOI: 10.1145/2897073.2897092	IEEE/ACM	Compare cross-platform approaches
S8	Heitkötter <i>et al.</i> (2013). Cross-platform model-driven development of mobile applications with md2. Proceedings of the 28th Annual ACM Symposium on Applied Computing-SAC 13. DOI: 10.1145/2480362.2480464	ACM	Introduce new approach
S9	Mercado <i>et al.</i> (2016). The impact of cross-platform development Computing-SAC 13. approaches for mobile applications from the users perspective. Proceedings of the International Workshop on App Market Analytics-WAMA 2016. DOI: 10.1145/2993259.2993268	ACM	Impact of using cross- platform
S10	Delia <i>et al.</i> (2015) "Multi-Platform Mobile Application Development Analysis." 2015, DOI: 978-1-4673-6630-4/15	IEEE	Compare cross-platform approaches
S11	Dalmaso <i>et al.</i> (2013). Survey, comparison and evaluation of cross platform mobile application development tools. 2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC). DOI: 10.1109/iwcmc.2013.6583580	IEEE	Compare cross-platform approaches
S12	Ali and Mesbah (2016). Mining and characterizing hybrid apps. Proceedings of the International Workshop on App Market Analytics-WAMA 2016. DOI: 10.1145/2993259.2993263	ACM	Compare cross-platform approaches
S13	Dunka <i>et al.</i> (2017). Hybrid mobile application based on ionic framework technologies. International Journal of Recent Advances in Multidisciplinary Research. 04. 3121-3130.	Google Scholar	Analyze cross-platform approach
S14	Vilcek and Jakopec (2017). Comparative analysis of tools for development of native and hybrid mobile applications. 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). DOI: 10.23919/mipro.2017.7973662	Google Scholar	Compare cross-platform approaches
S15	Majchrzak and Gr̄nli (2017). Comprehensive Analysis of Innovative Cross-Platform App Development Frameworks. Proceedings of the 50th Hawaii International Conference on System Sciences (2017). DOI:10.24251/hicss.2017.745	Google Scholar	Compare cross-platform approaches
S16	Angulo and Ferre (2014). A Case Study on Cross-Platform Development Frameworks for Mobile Applications and UX. Proceedings of the XV International Conference on	ACM	Compare cross-platform approaches

S17	Human Computer Interaction-Interaction 14. DOI:10.1145/2662253.2662280 Ali <i>et al.</i> (2017). Same App, Different App Stores: A Comparative Study. 2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft). DOI: 10.1109/mobilesoft.2017.3	IEEE/ACM	Compare cross-platform approaches
S18	VITOLS (2014) "Issues of Hybrid Mobile Application Development with PhoneGap: A Case Study of Insurance Mobile Application." International Baltic Conference, 2014.	Google Scholar	Cross-platform challenges and opportunities
S19	El-Kassas <i>et al.</i> (2014). ICPMD: Integrated cross-platform mobile development solution. 2014 9th International Conference on Computer Engineering and Systems (ICCES). DOI: 10.1109/icces.2014.7030977	IEEE	Introduce new approach
S20	Ahti <i>et al.</i> (2016). An Evaluation Framework for Cross-Platform Mobile App Development Tools. Proceedings of the 17th International Conference on Computer Systems and Technologies 2016 - CompSysTech 16. DOI: 10.1145/2983468.2983484	ACM	Analyze cross-platform approach
S21	Malavolta <i>et al.</i> (2015). End Users Perception of Hybrid Mobile Apps in the Google Play Store. 2015 IEEE International Conference on Mobile Services. DOI: 10.1109/mobserv.2015.14	IEEE	Compare cross-platform approaches
S22	Brucker and Herzberg (2016). On the Static Analysis of Hybrid Mobile Apps Lecture Notes in Computer Science Engineering Secure Software and Systems, 72-88. DOI: 10.1007/978-3-319-30806-7_5	Google Scholar	Introduce new approach
S23	El-Kassas <i>et al.</i> (2017). Taxonomy of Cross-Platform Mobile Applications Development Approaches. Ain Shams Engineering Journal, 8(2), 163-190. DOI: 10.1016/j.asej.2015.08.004	Google Scholar	Analyze cross-platform approach
S24	El-Kassas <i>et al.</i> (2016). Enhanced Code Conversion Approach for the Integrated Cross-Platform Mobile Development (ICPMD). IEEE Transactions on Software Engineering, 42(11), 1036-1053. DOI: 10.1109/tse.2016.2543223	IEEE	Introduce new approach
S25	Ciman and Gaggi (2017). An empirical analysis of energy consumption of cross-platform frameworks for mobile development. Pervasive and Mobile Computing, 39, 214-230. DOI: 10.1016/j.pmcj.2016.10.004	Google Scholar	Compare cross-platform approaches
S26	Biörn-Hansen <i>et al.</i> (2017). Baseline Requirements for Comparative Research on Cross-Platform Mobile Development.	Google Scholar	Analyze cross-platform approach
S27	Byalik <i>et al.</i> (2015). Native-2-native: Automated cross-platform code synthesis from web-based programming resources. ACM SIGPLAN Notices, 51(3), 99-108. DOI: 10.1145/2936314.2814210	ACM	Introduce new approach
S28	Chadha <i>et al.</i> (2017). Facilitating the development of cross-platform software via automated code synthesis from web-based programming resources. Computer Languages, Systems and Structures, 48, 3-19. DOI: 10.1016/j.cl.2016.08.005	ACM	Introduce new approach
S29	Tung <i>et al.</i> (2018). Cross-Platform Support for Rapid Development of Mobile Acoustic Sensing Applications. Proceedings of the 16th Annual International Conference on Mobile Systems, Applications and Services - MobiSys 18. DOI: 10.1145/3210240.3210312	ACM	Introduce new approach
S30	Boushehrinejadmoradi <i>et al.</i> (2015). Testing Cross-Platform Mobile App Development Frameworks (T). 2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE). DOI:10.1109/ase.2015.21	IEEE/ACM	Introduce new approach

Appendix B

Search String Queries

Search Query	Library	Initial Results	Refined Results (Most Relevant)	Filtration (Phase1)	Filtration (Phase2)
("Document Title":cross-platform) OR ("Document Title":hybrid) AND ("Document Title":mobile) AND ("Document Title":applications) OR ("Document Title":apps)	IEEE	2736	105	23	7
acmdlTitle:(("cross-platform" OR "hybrid") AND ("mobile") AND ("applications" OR "apps"))	ACM	200,080	241	25	13
("cross-platform" OR "hybrid") AND ("mobile") AND ("applications" OR "apps")	Google Scholar	383,000	9300	16	10