

# Analysis and Testing of Mobile Application Interfaces Implemented in UI frameworks

Uroosa kakepoto, Mohsin Ali memon, Sania bhatti

**Abstract**— Recently Mobile applications are progressively appreciated after being created in a cross-platform application development framework. In these types of frameworks, an application can be developed with on-time coding and can run on all different platforms. Irrespective of many advanced researches on these approaches, results are usually not satisfying. These frameworks are prone to long-winded customization and development endeavor that are significant and that is a very difficult situation for mobile application vendors and developers. Then in these types of cases, whether it would be a web application (implemented using web browsers) or native applications (implemented for every separate platform) are adopted. Recent developments have introduced new techniques. In this research paper, we have closely studied three cross-platform frameworks, namely React native, PhoneGap, and Ionic.

We have presented an extensive evaluation of the three frameworks.

**Index Terms**— mobile applications, cross platform frameworks interfaces, ionic, PhoneGap, React native

## 1 INTRODUCTION

Technological expansion has always had an impact on people like and organizations work sale. Mobile and network technologies are no exception. The tech devices used nowadays such as smart phones, watches and tablets are so ubiquitous they have dramatically altered the way people and companies/organizations do business, access data, interact with each other and do other things. Naturally, the role of mobile applications (simply mob apps) in this change immense. According to recent statics, Google play has about 2.5 million applications, Apple store has 1.8 million apps, and other including Windows and Amazon AppStore have about 1 million apps as of October 2019 [5]. Obviously, these figures are steadily increasing as a result of increasing consumer demand and advanced features new mobile phones have to offer. Now a day's mobile application vendors face the issue that they typically need to develop the mobile application for multiple platforms such as iOS, Android and windows at a time, as there are business potentials cannot be overlooked in each app market. One of the major challenges for an organization and vendors is that they need to offer consistent user experience across all platforms (operating systems), expecting unique platform / device features. One solution of this issue is to have a different platform-based application and separate development teams for each platform for the same application because each platform needs understanding and technology experience. This method, however, is the most expensive solution for Application development, updating and support. To survive in today's highly competitive software industry, software vendors/organizations need an alternate solution that helps in reducing development cost and resources developing the applications for all platforms in less amount of time. To meet this particular need, Cross platform mobile application frameworks are used to

help organizations/ developers to build the apps once with a similar code and run them on multiple platforms. To enhance the awareness of trending cross platform development and to provide the understanding on existent techniques, we have evaluated three frameworks. React native [9], Ionic framework [10], and PhoneGap [11] are the one mostly being used now a days. Usually practitioners and hobbyists discuss these frameworks actively. In this research paper, we attempt to provide business related direction to developers by evaluating the frameworks by using a real-life scenario focusing on user experience.

Although evaluation of other cross-platform frameworks is not so rare, some papers define a practice-oriented evaluation and few assess frameworks used by new generation. Such three frameworks are peculiarly attractive for analysis as they represent new steps of approaches which also presents paradigmatic shifts. Although our approach is design oriented, to strengthen our findings we have combined it with a survey (evaluation).

This research paper provides the following contribution:

To begin with, it gives an introduction of three innovative cross-platform frameworks. Second, it presents development of a prototype application, which ought to demonstrate valuable for assessment beyond our work. Third, it generalizes findings and comprises advice by conducting an evaluation of prototype app through usability testing. The remaining portion is organized as follows:

In section 2 we are presenting the related work. In Section 3 We have provided a general introduction of the frameworks used in our research. In section 4 we have described the design and implementation process of prototype applications. In section 5 evaluation of prototype applications through performance testing and usability testing. In Section 6

we have provided the finding and results of evaluation and Section 7 provides a conclusion of our research.

## 2 RELATED WORK

A bunch of different articles and other course readings also address app development topics. At least a few of these are covering cross-platform strategies, despite the fact that not usually as the main topic. To shed light on related papers we look on that research, which compares several cross-platform application development frameworks. Here, we don't include papers that examine methods and technology for web application development. Web technology offers one means of creating platform-wide apps [6], [7]; however, creating web apps can be seen as an alternative to using a cross-platform framework [8], and mobile apps can be used as a benchmark for assessment. Moreover, we don't include papers here that handle native application development for more than one platform. Native apps are also useful for benchmarking cross-platform approaches, particularly concerning their UI and their performance. An outline of papers that have compared cross-platform development system is given in Table 1. A few other works, other than comparative papers, can be seen as preconditions for our evaluation, including the specific challenges of application enhancement [16]. Where appropriate, works specifically similar to our assessment perspective, that are to be cited in the remainder of this article.

Table 1  
Summary of Related Work:

Paper	Year	Framework/Approach	Particularities
[01]	2018	PhoneGap, xamarin	Mostly focused on Developers perspective
[02]	2017	Cross compilation, Virtual machine, Web based approach	Compare each of the Approach for mobile Application Development
[12]	2016	Xamarin, Cordova/ PhoneGap	Evaluation based on a Prototype Application
[13]	2016	Adobe PhoneGap	Differentiation between Phone Gap and Native Development approach
[17]	2015	PhoneGap, Appcelerator, Xamarin, Smart face App Studio and Titanium	Comparison with respect to development issues
[14]	2013	Sencha Touch, Phone Gap, Titanium	The performance is examined in terms of CPU usage memory and power consumption
[15]	2012	Rhodes, PhoneGap, Dragon Rod, mosync	Focuses on features of frameworks with respect to market needs

A lot of specialized papers have also been published. An example of this is a specific energy consumption center for applications created with cross-platform frameworks [17]. Additionally, common evaluations of the various platforms [18] must be taken into account. In addition, Huy and van Thanh [19] proposed criteria for evaluation of applications. They don't provide a real evaluation, they suggest "how to do

it." Their thinking is to require distinctive perspectives, particularly those of developers, customers and service providers.

## 3 CROSS-PLATFORM MOBILE APPLICATION DEVELOPMENT TOOLS

In this research, we basically consider three prevalent and esteemed cross-platform mobile app development tools/frameworks used by the developers (professional/hobbyist). First is PhoneGap, second one is Ionic and third is React native, all of them are used in the category of cross-platform native app development approach.

### 3.1 React Native

React Native is a cross-platform, JavaScript based framework. It was developed by facebook community. This allows developers to build improved and native-like apps using a single code base. The purpose of this framework is to use JavaScript and React.js to create first-rate native application experiences. React native embraces the platform's native actions and behaviors and thus offers flawless native experience. The advantage here is that, with the benefit of native UI elements, it brings complete focus to user interfaces.

### 3.2 PhoneGap

PhoneGap is an open source framework for development of hybrid mobile applications. It was originally developed by Nitobi and then the company was acquired by Adobe in 2011. PhoneGap uses standard and well-known web technologies such as HTML, CSS, and JavaScript to create cross-platform mobile apps without using native development languages. It packages app code into an executable program that can run on an array of mobile devices. Developers can write code once and deploy their app across multiple mobile operating systems such as iOS, Android, Windows Phone 8, BlackBerry, and Amazon FireOS. PhoneGap provides a JavaScript programming interface that allows developers to access platform-specific features with plain JavaScript [4].

### 3.3 Ionic

Ionic was developed by Drifty.co, is a standard hybrid mobile app development framework. It enables web developers to use a common code base to build apps for different platforms. Ionic is an open source platform offering native look and feel tools and services to develop mobile UI. Ionic framework needs a native wrapper in order to run on mobile devices. Since Ionic is built on top of AngularJS and Apache Cordova, you'll need basic knowledge of these technologies. You need to learn HTML, CSS and JavaScript.

## 4 DESIGN AND IMPLEMENTATION

This section explains the design and implementation of the prototype applications used for the evaluation of the selected frameworks. It comprises of two parts, in first part, we describe the implementation design of the two interfaces that are to be developed, whereas, in second part, we describe the implementation of the interfaces that are being designed using

the above selected frameworks. By using each of three frameworks, three different applications are developed for evaluation.

#### 4.1 Design and implementation (of prototype 1)

We have designed a prototype application using the cross-platform frameworks to evaluate them from different perspectives. The main interface of our developed prototype is shown in Fig:1. This application is a women safety application, which can be used by women in emergency situations. It is developed by all three frameworks i.e. React native, Ionic and PhoneGap.

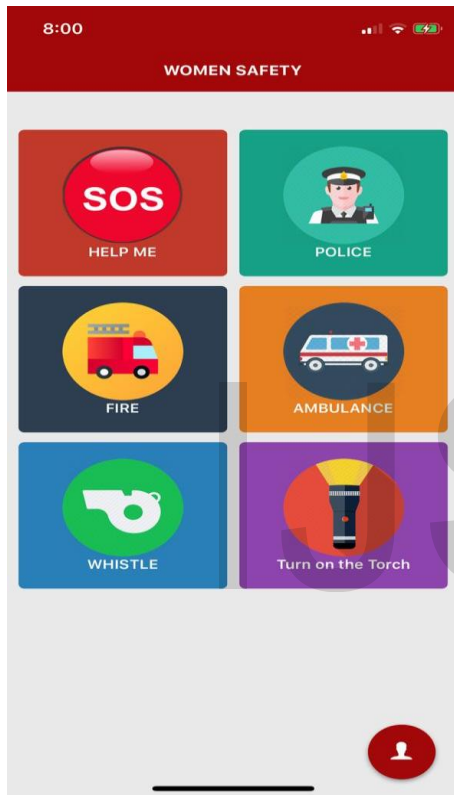


Fig: 1: Main interface of prototype application

Fig: 1 shows the main interface of application. This application has many different features:

1. SOS button will send the exact location and an Emergency message to saved emergency contact.
2. Police, Fire and Ambulance numbers allow to call their respective numbers.
3. Whistle sets an alarm.
4. And turn on the torch will turn on mobile light.

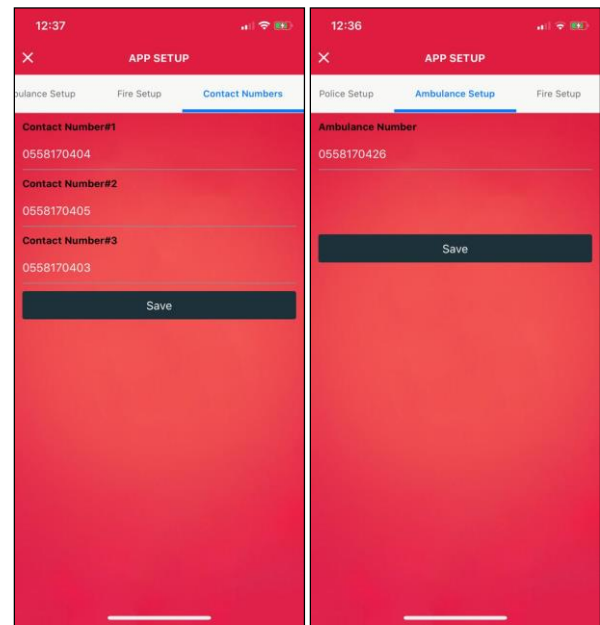


Fig: 2: Women Safety app interfaces

#### 4.2 Design and implementation (of prototype 2)

We have developed another prototype application using the selected cross-platform frameworks i.e. React native, Ionic and PhoneGap. This application is an online shopping application. Fig: 3 shows the main interface of the developed application.

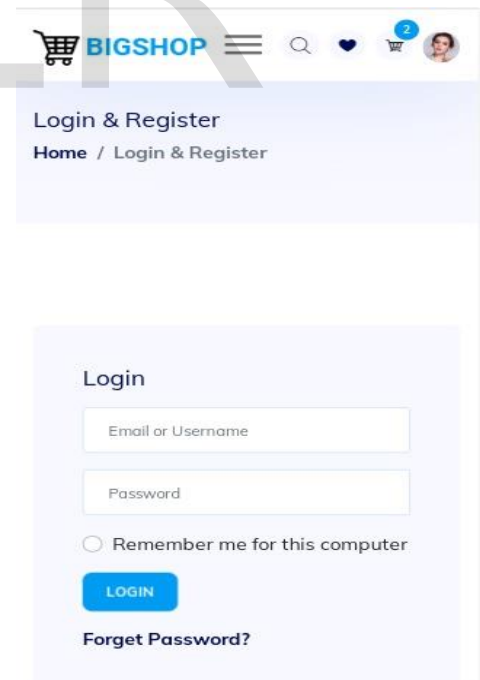


Fig: 3: Main interface of online shopping application

Now a days, ecommerce and online shopping applications are one of the important means of business. Because of its accessibility and convenience, more people prefer online shopping. The shopping application is consisting of login,

home, items, orderlist, wishlist and checkout interfaces. Some of them are depicted in Fig: 4 and Fig: 5.

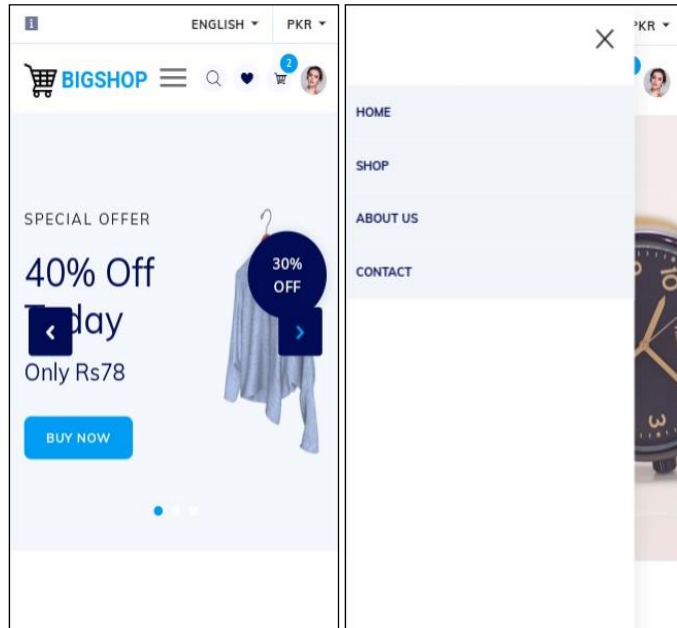


Fig: 4: Home and menu list interfaces of prototype 2

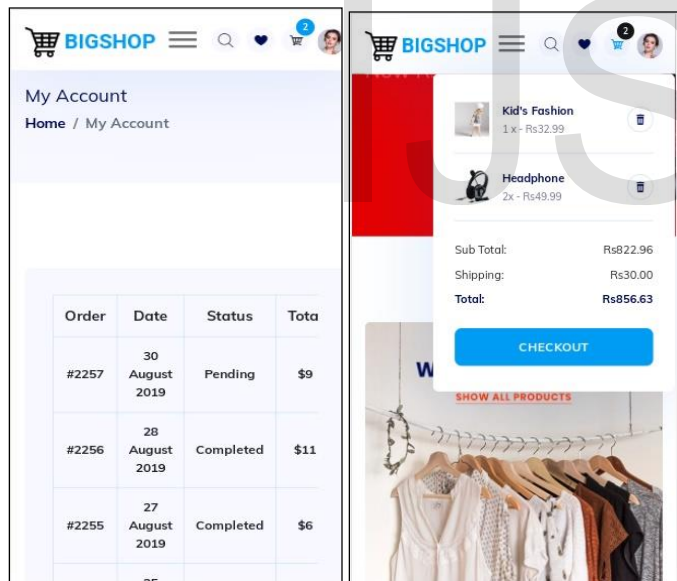


Fig: 5: Interfaces of checkout and orderlist pages.

## 5. EVALUATION

The evaluation of the three frameworks has been performed by deploying two different approaches; one by measuring four performance parameters and other by collecting evaluation feedback from users.

### 5.1 Evaluation of Performance Benchmarks

Performance is one of the most important factor to be considered while assessing the cross-platform frameworks. These are the four factors which are evaluated for analyzing the performance of the application which are CPU, memory, power

usage and Application size.

### 5.1.1 Performance evaluation of three frameworks

#### ( Prototype 1 ):

We developed a women safety application in the selected frameworks. They are then examined with the above performance aspects. Table 2 shows the performance criteria of women safety app.

Table 2  
Performance Criteria Of Women Safety App

Criteria	React Native	Ionic	PhoneGap
CPU usage	13.5 %	18.7%	17.1%
Memory Usage	760 MB	1126 MB	1202 MB
Battery Usage	412.2 mW	455 mW	482 mW
Application size	9.2 MB	4.0 MB	3.6 MB

Performance evaluation shows that the prototype developed using React native has better CPU, memory and battery usage than ionic /PhoneGap but its application size is bigger than other two

### 5.1.2 Performance evaluation (Prototype 2):

These are the performance criteria for the other prototype application (online shopping app) presented in table 3.

Table 3  
Performance Criteria Of Online Shopping Application

Criteria	React Native	Ionic	PhoneGap
CPU usage	12.8 %	21.3%	20.1%
Memory Usage	868 MB	1049 MB	988 MB
Battery Usage	476.2 mW	522 mW	509.5 mW
Application size	8.7 MB	3.4 MB	3.9 MB

Performance evaluation also shows that the prototype 2 developed using React native has better CPU, memory and battery usage than ionic /PhoneGap and its application size is more.

### 5.2 Evaluation of Prototype Applications:

The evaluation was carried out in order to compare and find the best framework among the three selected frameworks. During the evaluation process, 50 different users were asked to test all the developed prototype applications. Among the participants who took part in evaluation were between the age of 19 and 28. This evaluation was presented to find the best application by general understanding of the students through usability testing. The questions used for evaluation and the participant's responses are given in Table 4. The results and users feedback regarding the use of applications of both prototype systems are calculated and presented in graphs in section 6.

## 6 RESULTS AND FINDINGS OF EVALUATIONS:

While conducting the evaluation, the participants were given options from "strongly agree" to "strongly disagree" to select the option best suited for them. This evaluation was conducted by all participants who performed the usability testing of the applications Fig: 5, 6, 7, 8 9, 10, 11 shows the result of the



evaluation.

**Table 4**  
Questions and responses included in evaluation

Questions	Strongly agree	Agree	Disagree	Strongly Disagree
1. Application developed in React native is "Easy to use".	54%	46%		
2. Application developed in Ionic is "Easy to use".	50%	50%		
3. Application developed in PhoneGap is "Easy to use".	45%	55%		
4. I like the User Interface for the React native application.	45%	55%		
5. I like the User Interface of Ionic application.	42%	58%		
6. I like the User Interface for the PhoneGap application.	41%	59%		
7. I think the designs and working of the icons/buttons in React native applications.	52%	48%		
8. I think the designs and working of the icons/buttons in Ionic applications.	50%	50%		
9. I think the designs and working of the icons/buttons in PhoneGap applications.	48%	52%		
10. The content is not visible and legible in the React native applications.			2%	98%
11. The content is not visible and legible in the Ionic applications.			4%	96%
12. The content is not visible and legible in the PhoneGap applications.			4%	96%
13. I think the loading speed of the React native mobile application is fast.	53%	47%		
14. I think the loading speed of the Ionic	49%	51%		

mobile application is fast.				
15. I think the loading speed of the PhoneGap mobile application is fast.	48%	52%		
16. I think the Interface design of the React native application is visually appealing	55%	45%		
17. I think the Interface design of the Ionic Application is Visually appealing	50%	50%		
18. I think the Interface design of the PhoneGap application is Visually appealing	49%	51%		
19. The overall experience of using the React native mobile application was satisfactory.	59%	41%		
20. The overall experience of using the Ionic mobile application was satisfactory.	53%	47%		
21. The overall experience of using the PhoneGap mobile application was satisfactory.	50%	50%		

Fig 6 is showing the combined comparative analysis of question 1, 2, 3. This question was related to "Ease of Use" of application. During testing of React native application, out of 50 students, 54% of students replied with "strongly agree" after using the application and remaining 46% replied with "agree". None of the user has found it difficult while using the React native application. In testing the Ionic application, 25 users among total of 50 users responded with "strongly agree", while the remaining 25 users replied with "agree". None of the user has found it difficult while using the application. In testing the application developed in PhoneGap, among all users, 45% responded "strongly agree" and 55% responded with "agree" after using this application. After comparative analysis of results, all interfaces were found easy to use and among all developed prototypes, Applications developed using react native were rated better than other two frameworks.

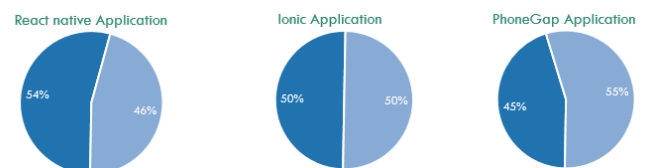


Fig: 6: Evaluation results of Application

Fig 7 is showing the combined comparative analysis of question 4, 5, 6. These questions are related to user interface (UI) of the applications. For example, design, colors used in the application.

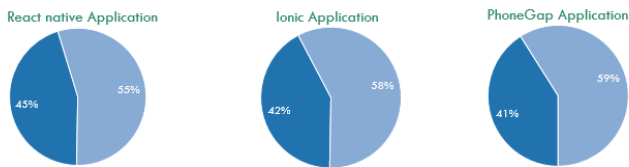


Fig 7: Evaluation results of Application

Fig 8 is depicting the combined comparative analysis of question 7, 8, 9. These questions are about the appearance and working of the icons/buttons used in the applications (like color, design). It is perceived that all of students provided positive results.

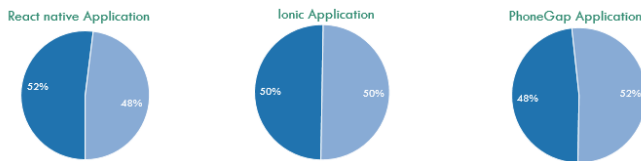


Fig 8: Evaluation results of Application.

Fig 9 is portraying the combined comparative analysis of question 10, 11, 12. These questions are about the understanding of content of the applications, and all students provided positive results.

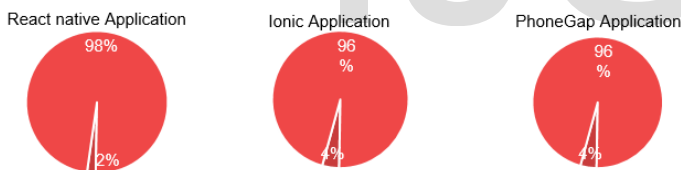


Fig 9: Evaluation results of Applications

Fig 10 is showing the combined comparative analysis of question 13, 14, 15. These questions are about the loading speed of application (Page loading, working).

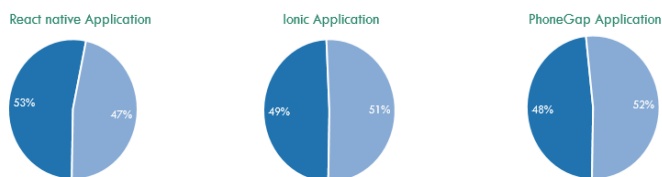


Fig 10: Evaluation results of Applications.

Fig 11 provide the evaluation results of questions 16, 17, 18. These questions are about the design of application (visual, colors, layouts structure).

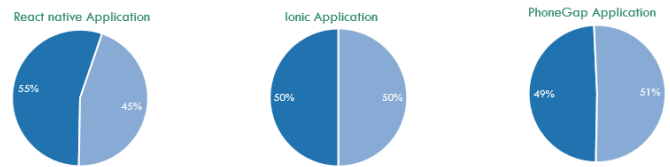


Fig 11: Evaluation results of Applications.

Fig 12 represents the analysis 19, 20, 21. These questions represents the overall preference of the users for the application (satisfaction, likeness, and understanding).

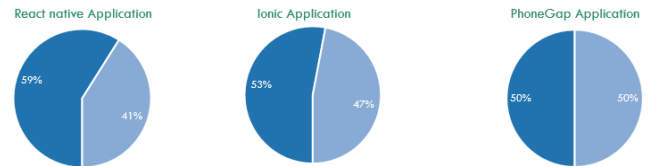


Fig 12: Evaluation results of Applications.

## 7 CONCLUSION:

This paper presents the comprehensive evaluation of the three selected frameworks i.e. React native, Ionic and PhoneGap through two perspectives which are performance measures and usability. After evaluating performance measures it is not straight forward to decide which framework is better option for developers. Despite their common objective, they all have distinctive features that make each one superior and preferable to the others. Therefore, developers /organizations need to understand the advantages and disadvantages of them, evaluate each one considering their own specific development requirements and constraints, and then making their choices wisely. If a developer wants an excellent user experience, convenience, and reduced costs of cross-platform development then React native is better choice, however it will take more time in development than ionic/PhoneGap. But if a developer wants to quickly develop an application or convert a website into a mobile app without high UX.

## REFERENCES

- [1] J. Goetz and Y. Li, "Evaluation of Cross-Platform Frameworks for Mobile Applications". *International Journal of Engineering and Innovative Technology (IJEIT)* Vol 8, Issue 3, pp. 10-17, sep. 2018
- [2] W. S. El-Kassas, B. A.Abdullah, A. H.Yousef and A. M.Wahba, "Taxonomy of cross-platform mobile applications development approaches" *Ain Shams Engineering Journal* Vol 8, Issue 2, pp. 163-190, June 2017,
- [3] M.Popa, "Considerations Regarding the Cross-Platform Mobile Application Development Process". *Economy Informatics* vol. 13, Jan. 2013
- [4] R. Purusothaman and G. Natili ".PhoneGap Beginner's Guide" (3rd ed.). Birmingham, UK, 2015, Packt Publishing ltd.
- [5] n.d "The Statista website" [Online]. Available:

<https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/> 2019

[6] A. Connors and S. Bryan "Mobile web application best practices." [Online] Available: <http://www.w3.org/TR/2010/REC-mwabp-20101214> (2010).

[7] H. Heitkötter, T. A. Majchrzak, B. Ruland, and T. Weber, "Comparison of Mobile Web Frameworks," in *LNBIP*, vol. 189. Springer, pp. 119–137, July 2014.

[8] A. Charland and B. Leroux, "Mobile application development: web vs. native," *Communications of the ACM*, vol. 54, pp. 49–53, May 2011.

[9] React native community "the facebook github website" [online] Available: <https://facebook.github.io/react-native/docs/getting-started> 2019

[10] Ionic community "The Ionic framework website" [online] Available: <https://ionicframework.com/community> 2019

[11] Apache software development "The PhoneGap website" [online] Available: <http://docs.phonegap.com/> 2019

[12] L. Öberg, "Evaluation of Cross-Platform Mobile Development Tools Development of an Evaluation Framework", Dissertation, 2016.

[13] V. Ahti, S. Hyrynsalmi, and O. Nevalainen. "An Evaluation Framework for Cross-Platform Mobile App Development Tools: A case analysis of Adobe PhoneGap framework." *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*, vol 16, pp. 41-48, June 2016.

[14] I. Dalmaso, S. K. Datta, C. Bonnet and N. Nikaiein, "Survey, comparison and evaluation of cross platform mobile application development tools," *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Sardinia, pp. 323-328, July 2013.

[15] M. Palmieri, I. Singh and A. Cicchetti, "Comparison of cross-platform mobile development tools," *2012 16th International Conference on Intelligence in Next Generation Networks*, Berlin, pp. 179-186, Oct. 2012.

[16] Raj, CP Rahul, and Seshu Babu Tolety. "A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach." *India Conference (INDICON), 2012 Annual IEEE*. IEEE, pp.625-629, Dec. 2012.

[17] Tunali, V., and S. Zafer. "Comparison of Popular Cross-Platform Mobile Application Development Tools", Celal Bayar University: Faculty of Technology Department of Software Engineering, Maltepe University: Faculty of Engineering and Natural Sciences Department of Software Engineering (2015).

[18] M. E. Joorabchi, A. Mesbah, and P. Kruchten, "Real challenges in mobile app development," *ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. IEEE, pp. 15–24, Oct 2013.

[19] M. Ciman and O. Gagg, "Measuring energy consumption of cross-platform frameworks for mobile applications," *LNBIP*, vol. 226, pp. 331–346, Dec. 2015.

[20] T. M. Grønli, J. Hansen, G. Ghinea, and M. Younas, "Mobile application

platform heterogeneity: Android vs Windows Phone vs iOS vs Firefox OS," *IEEE 28th International Conference on Advanced Information Networking and Applications*, pp. 635–641, May 2014.

[21] N. P. Huy and D. vanThanh, "Evaluation of mobile app paradigms," *Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia*, pp. 25–30, Dec 2012.

[22] C. Chedeau. "From Hackathon to React Native" Online video, on infoQ website [online] Available: [https://www.infoq.com/presentations/react-native-facebook\\_2019](https://www.infoq.com/presentations/react-native-facebook_2019)

## Biographies :

**Uroosa Kakepoto** is a M. E student of Software Engineering in Mehran University of Engineering and Technology, Jamshoro, Sindh, Pakistan. She obtained her B.E (Software Engineering) from Mehran University of Engineering and Technology, Jamshoro, Sindh, Pakistan. Her research area is evaluation of Mobile Application Development frameworks.

**Mohsin Ali Memon** is working with the department of Software Engineering, Mehran UET, Jamshoro, Sindh, Pakistan. He obtained his PhD degree from the Department of Computer Science, University of Tsukuba in 2014. His research interests include interaction technologies, life logging, privacy control methods and machine learning. He received his B.Eng. in Software Engineering and M.Eng. in Information Technology from Mehran University of Engineering and Technology, Pakistan, in 2006 and 2009, respectively.

**Sania Bhatti** is working with the Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro Sindh, Pakistan. She obtained her PhD from the University of Leeds, the United Kingdom in 2010 under the scholarship of the faculty development program. Her research interests include modelling, simulation, communication networks and machine learning algorithms. She has published more than twenty national and international Journal papers and various international conference papers. She was awarded with two research grant by Microsoft in the field of Artificial Intelligence as principle investigator.