sustainability

MDPI

Article

Towards Accessibility and Inclusion of Native Mobile Applications Available for Ecuador in Google Play Store

Patricia Acosta-Vargas [1,2,*], Sebastian Zarate-Estrella [2], Franccesca Mantilla-Vaca [2], Sylvia Novillo-Villegas [2], Christian Chimbo [2] and Sergio Luján-Mora [3]

1 Intelligent and Interactive Systems Laboratory, Universidad de Las Américas, Quito 170125, Ecuador
2 Facultad de Ingeniería y Ciencias Aplicadas, Carrera de Ingeniería en Producción Industrial, Universidad de Las Américas, Quito 170125, Ecuador; sebastian.zarate@udla.edu.ec (S.Z.-E.); franccesca.mantilla@udla.edu.ec (F.M.-V.); sylvia.novillo@udla.edu.ec (S.N.-V.); christian.chimbo@udla.edu.ec (C.C.)
3 Department of Software and Computing Systems, University of Alicante, 03690 Alicante, Spain; sergio.lujan@ua.es
* Correspondence: patricia.acosta@udla.edu.ec

Abstract: This article aims to evaluate the level of compliance with the accessibility requirements of the most popular native Android mobile applications, for which a sample of 50 Google Play Store applications available in Ecuador was taken. A five-phase method using the Accessibility Scanner tool was used to evaluate the apps. The results revealed that 47.5% are related to problems with tactile orientation, followed by the labeling of elements with 28.2%, and text contrast with 9.2%. The highest number of barriers found in the evaluation of mobile applications corresponds to the principle of operability with 53.9%. This study reveals that, although social networks are widely used, they have 28.7% of accessibility problems. Basing accessibility analysis exclusively on an automatic tool is very limited since it neither detects all errors nor are the errors they detect accurate. However, we suggest complementing the automatic review evaluations with a manual method based on heuristics to ensure an adequate level of accessibility in mobile apps. In addition, we recommend using this study as a starting point to create a software tool using WCAG 2.1 based on artificial intelligence algorithms to help developers evaluate accessibility in mobile apps.

Keywords: Accessibility Scanner; assessment; inclusion; method; native mobile application; sustainability; WCAG 2.1

check for updates

Citation: Acosta-Vargas, P.; Zarate-Estrella, S.; Mantilla-Vaca, F.; Novillo-Villegas, S.; Chimbo, C.; Luján-Mora, S. Towards Accessibility and Inclusion of Native Mobile Applications Available for Ecuador in Google Play Store. *Sustainability* **2021**, *13*, 11237. https://doi.org/10.3390/su132011237

Academic Editors: Pablo Rodríguez-Gonzálvez, Eloy López Meneses, Esteban Vázquez-Cano and María Elena Parra-González

Received: 21 July 2021
Accepted: 8 October 2021
Published: 12 October 2021

1. Introduction

Today, almost everything we do in our daily lives is supported by technology. Technology has passed through all levels of society. One of the most significant changes introduced is facilitating interpersonal communication and quick and instant access to information. Likewise, mobile devices have revolutionized our lives, and with them, native mobile applications have become necessary tools to use various services without leaving home.

Information and communication technologies are key factors [1] in improving well-being and integration into society, especially for people with disabilities and chronic diseases. To this end, new tools and mobile applications should be made available to help these people in their daily lives.

Furthermore, the World Health Organization (WHO) [2] and the World Bank [3] indicate that 15% of the world's population experience some disability. Between 110 million and 190 million people, or one-fifth of the world's population, are affected by significant disabilities.

Inclusion of people with disabilities and expanding equality of opportunity are central to the World Bank's work to create sustainable and inclusive communities [4], in line with corporate objectives to seek mechanisms to eradicate extreme poverty and promote shared prosperity.

Several applications on the web provide services to millions of users; there are mobile applications as a strategy to complement these services. These applications called "apps" have grown exponentially, there are about 9 million mobile applications in Google Play Store [5], but not all of them have an accessible and inclusive design.

Therefore, in this research, we suggest applying the Web Content Accessibility Guidelines (WCAG) 2.1 [6] from the World Wide Web Consortium, including a wide range of recommendations to make application content more accessible. WCAG is based on the four accessibility principles: (1) perceivable, (2) operable, (3) understandable, and (4) robust. In addition, WCAG includes 13 guidelines and 78 conformance criteria, plus an unspecified number of sufficient techniques and advisory techniques.

WCAG 2.1 helps make content more accessible to all people, including accommodations for blindness and low vision, deafness and hearing loss, limited movement, speech disabilities, photosensitivity, and some accommodations for learning and cognitive disabilities. The implementation of WCAG 2.1 [6] and Mobile Accessibility [7] make the content of mobile applications more useful for all users, especially people with disabilities.

This research evaluates free applications from the Google Play [8] top 50 Apps hit list for native Android applications. After evaluating the top 50 apps using Accessibility Scanner, the results are analyzed to offer proposals for improving accessibility in native mobile apps. This automatic review tool does not replace a manual evaluation or guarantee that an application is accessible.

The automatic review method comprises five phases: (1) Define the applications to be evaluated, (2) define the test scenario, (3) evaluate each application in Accessibility Scanner, (4) classify and analyze the results, and (5) record the results.

Following this introduction, Section 2 presents a review of the literature that justifies the research problem. In Section 3, the proposed methodology is explained. The results obtained are presented in Section 4. Discussion of the results is in Section 5, and finally, conclusions and possible future work are given in Section 6.

2. Literature Review

Nowadays, mobile applications are essential. The study [9] argues that the existence and use of social media on the websites of parliaments, according to the survey conducted in 2016, use Facebook, Twitter, YouTube, and video streaming. The study shows that social networks improve communication between parliaments and citizens.

According to Statista statistics [10], in the first quarter of 2021, Android users could choose from 3.48 million apps, making Google Play the app store with the most significant number of apps available. Apple App Store was the second-largest app store, with approximately 2.22 million apps available for iOS.

Despite having millions of mobile apps that help solve many problems in our daily lives, not all of them are accessible. For a mobile application to be inclusive, it must meet the WCAG 2.1 guidelines, for which we suggest evaluating the application with a combined method throughout the entire software development cycle [11].

Our previous study [12] indicates that the use of mobile devices has grown exponentially; in this context, mobile devices are intended to democratize access to knowledge on different topics; however, accessibility standards have been neglected. The results obtained from the scoping review suggest applying WCAG 2.1 in mobile applications to achieve adequate accessibility.

In the literature review, we found several studies [11,13,14] using manual and automatic review methods related to the evaluation of mobile applications in which the Accessibility Scanner was applied.

The author's research [15] evaluated the accessibility status in mobile apps related to GUI structures and conformance to accessibility guidelines of 479 Android apps from 23 business categories. The results showed that accessibility issues were mainly due to lack of element focus, lack of element description, poor text color contrast, lack of sufficient spacing between elements, and lack of good element quality.

The study [16] on mobile app accessibility barriers highlights that a wide variety of factors influence the accessibility of an app and presents the analysis on a large scale. Seven accessibility barriers are highlighted: few focusable elements with TalkBack, missing labels, duplicate labels, uninformative labels, editable TextViews. The study results argue that the current state of accessibility on Android suggests improvements for the app ecosystem and demonstrate analysis techniques that can be applied in future app-accessibility evaluations.

The study [17] addresses the case of mobile applications, where accessibility is achieved through existing features on mobile devices, such as TalkBack on Android. After a qualitative analysis, the results propose lessons to guide future research to help developers support users in need of assistive features.

The research [18] argues that there are several accessibility features available on mobile platforms that describe user interface components. In addition, the paper indicates that they trained a robust, memory-efficient model on the device to detect user interface elements. It argues that they created a screen recognition for VoiceOver. The study results were conducted with nine users with screen readers and validated with a focus on improving the accessibility of mobile applications.

The research [19] argues that there are accessibility issues in mobile applications. It presents a checklist to assess whether applications have prevented social exclusion and facilitated access to information for many users, including citizens with disabilities. The study concludes that, despite the great potential shown by mobile applications, several accessibility issues have not been adequately addressed.

The author's study [20] argues that designing accessible mobile applications for people with disabilities requires appropriate evaluation techniques and tools to identify accessibility issues. Furthermore, automated accessibility evaluation tools can support evaluation tasks and make evaluators more productive in performing repetitive analysis.

Our previous studies by the authors [21,22] refer to a random sample of mobile applications evaluated with the accessibility considerations proposed in WCAG 2.1. Another previous study by the authors [23] proposes a heuristic method based on Brajnik's barrier walk [24]. The method includes some improvements to evaluate accessibility in mobile applications based on WCAG 2.1. The results of this research can help future work on the development of accessible mobile applications.

3. Materials and Methods

According to Statista statistics [10], Android users were able to choose among 3.48 million applications from which the 50 most popular free apps were selected based on their downloads and ratings assigned by users for Android available in Ecuador. The reason for applying this last filter is geolocation, which refers to the ability to obtain the actual geographic location of an object, such as a cell phone or a computer connected to the Internet.

The accessibility evaluation method for native mobile apps is summarized in five steps detailed in Figure 1. The automatic accessibility validator for mobile apps used in this evaluation process is the Accessibility Scanner. This validator does not guarantee the accessibility of the application; it serves as a guide for application developers to identify the main accessibility barriers related to WCAG 2.1 in a faster way. A complete evaluation is achieved by applying a combined method with an automatic validator and a manual evaluation.

In this research, we applied an automatic review method to evaluate accessibility in native mobile applications under the Android operating system comprising:

(1) Define the applications to be evaluated; in this step, we selected the 50 most successful free applications from Google Play for Android available in Ecuador. The sample was analyzed from 19 April to 28 June 2021. Table 1 contains the mobile applications evaluated in this study. They are sorted first by the number of downloads and the apps rated by Google Play for Android users, app type, and age group.

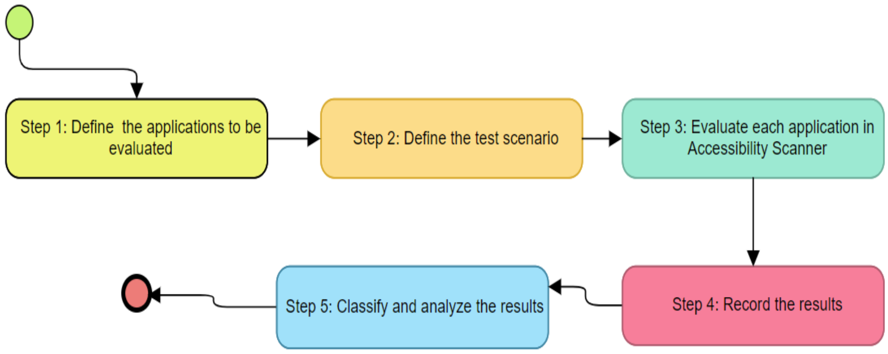

Figure 1. Accessibility Evaluation Method for Mobile Applications.

Table 1. Mobile applications were evaluated in this study.

Id	Mobile Applications	Download	Application Type	Age	Ranking
1	WhatsApp	5000 M	Communications	All	4.2
2	Google Maps	5000 M	Travel	All	4.2
3	Facebook	5000 M	Social	Teenagers	4.1
4	YouTube	5000 M	Video applications	Teenagers	4.1
5	Candy Crush	1000 M	Casual	All	4.6
6	TikTok	1000 M	Social	Teenagers	4.5
7	Google Traductor	1000 M	Tool	All	4.5
8	Instagram	1000 M	Social	Teenagers	4.4
9	Netflix	1000 M	Entertainment	Teenagers	4.4
10	Skype	1000 M	Communications	All	4.3
11	Snapchat	1000 M	Social	Teenagers	4.3
12	Google Documents	1000 M	Productivity	All	4.3
13	Messenger	1000 M	Communications	All	4.2
14	Twitter	1000 M	Social	All	4.2
15	Wish	500 M	Shopping	Teenagers	4.6
16	Pinterest	500 M	Lifestyle	Teenagers	4.6
17	Telegram	500 M	Communications	All	4.5
18	Spotify	500 M	Music and audio	Teenagers	4.5
19	LinkedIn	500 M	Business	All	4.4
20	Uber	500 M	Food and drink	All	4.4
21	Shazam	500 M	Music and audio	Teenagers	4.4
22	Likee	500 M	Video applications	Teenagers	4.3
23	Zoom	500 M	Business	All	3.9
24	Free Market	100 M	Shopping	All	4.8
25	EBay	100 M	Shopping	Teenagers	4.7
26	Amazon	100 M	Shopping	Teenagers	4.6
27	Duolingo	100 M	Education	All	4.6
28	Microsoft Outlook	100 M	productivity	All	4.6

Table 1. *Cont.*

Id	Mobile Applications	Download	Application Type	Age	Ranking
29	Discord	100 M	Communications	Teenagers	4.5
30	CapCut	100 M	Video applications	All	4.5
31	Uber Eats	100 M	Food and drink	All	4.4
32	Twitch	100 M	Entertainment	Teenagers	4.4
33	Waze	100 M	Food and drink	All	4.4
34	Disney+	100 M	Entertainment	Teenagers	4.3
35	YouTube Kids	100 M	Music and audio	All	4.3
36	PayPal	100 M	Finance	All	4.2
37	Microsoft Teams	100 M	Company	All	4.1
38	Google Meet	100 M	Company	All	3.7
39	One Booster	50 M	Tool	All	4.7
40	Radio FM	50 M	Music and audio	Teenagers	4.6
41	Reddit	50 M	Social	All	4.5
42	Parchis	50 M	Game	Teenagers	4.3
43	Shop	10 M	Shopping	All	4.8
44	Kahoot!	10 M	Education	All	4.6
45	McDonald's App	10 M	Food and drink	All	4.6
46	Rappi	10 M	Food and drink	All	4.3
47	IRS2Go	10 M	Finance	All	3.9
48	Pedidos Ya	10 M	Food and drink	All	3.1
49	Pichincha Banca	1 M	Finance	All	4.5
50	Tipti	0.1 M	Shopping	All	3.4

Where M means million.

(2) Define the test scenario; in this step, we defined the test environment by running BlueStacks emulator version 4.240.20.1016 under Windows 10; we visited the first screen of each application and interacted with the available menu. This process was applied to the 50 mobile applications evaluated. As for our study, we considered the following: we selected only the first screen of the application because it is the most critical element in terms of accessibility. If the first screen is not accessible, users could have problems even reaching other mobile application screens.

(3) Evaluate each application in Accessibility Scanner; in this step, we ran BlueStacks emulator version 4.240.20.1016 under Windows 10 and the automatic review tool Accessibility Scanner. The evaluation of the mobile applications was conducted from 19 April to 28 June 2021.

(4) Record the results; in this step, we recorded the evaluation results with the Accessibility Scanner. The data were organized according to the number of downloads and the rating assigned by Google Play users. The results section shows the evaluation of accessibility in mobile applications. The dataset for replicating the experiment of accessibility evaluation in mobile applications is available in the Mendeley repository [25].

(5) Classify and analyze the results; based on the results obtained from the evaluation of the mobile applications with Accessibility Scanner, we organized the information to analyze the data recorded from the 50 applications analyzed. The details of the analysis performed are available in the Mendeley repository [25]. In the results and

discussion section, we will present the analysis and correlation performed on the applications evaluated in this study.

4. Results

This section presents the results of the evaluation of the 50 mobile applications; we proceeded to classify and tabulate each of the accessibility barriers identified by the Accessibility Scanner. The information was summarized using Microsoft Excel with the support of dynamic reports. The problems were classified according to the WCAG for mobile applications; the authors classified the Accessibility Scanner tests; Table 2 presents mobile accessibility and how WCAG 2.0 and other W3C/WAI guidelines apply to mobile [7].

Table 2. Mobile accessibility: how WCAG 2.0 and other W3C/WAI guidelines apply to mobile.

Features	Failure	%	Mobile Accessibility Considerations	Mobile Accessibility Considerations (Description)	Principle WCAG 2.1	WCAG 2.1 (Success Criterion)	WCAG 2.1 (Success Criterion) Description	Level
Text contrast	40	9.2	2.3	Contrast	Perceivable	1.4.3	Contrast (Minimum)	AA
Item descriptions	28	6.4	4.6	Provide instructions for custom touchscreen and device manipulation gestures	Perceivable	1.3.5	Identify Input Purpose	AA
Touch target	207	47.5	3.2	Touch Target Size and Spacing	Operable	2.5.5	Target Size	AAA
Element type label	13	3.0	2.1	Small Screen Size	Operable	2.5.3	Label in Name	A
Clickable elements	7	1.6	3.3	Touchscreen Gestures	Operable	2.5.1	Pointer Gestures	A
Image contrast	8	1.8	4.4	Grouping operable elements that perform the same action	Operable	2.4.4	Link Purpose (In Context)	A
Editable element tag	3	0.7	4.5	Provide clear indication that elements are actionable	Understandable	3.2.4	Consistent Identification	AA
Item label	123	28.2	4.2	Consistent Layout	Understandable	3.2.3	Consistent Navigation	AA
Type of element not supported	7	1.6	5.3	Support the characteristic properties of the platform	Robust	4.1.2	Name, Role, Value	A

Figure 2 shows the accessibility issues of the evaluated mobile applications; the issues are summarized in column E of Table 3. Figure 2 shows the percentage of accessibility errors in the applications evaluated; the most accessible is Shazam, followed by Google Documents, Microsoft Teams, Google Traductor, Candy Crush, Zoom, McDonald's App, Twitter, Google Meet, Discord, and Google Maps. The applications with the most accessibility issues are Reddit with 11.2%, followed by TikTok with 7.3%, Twitch with 6.2%, Wish with 5.3%, Snapchat with 4.4%, Netflix and LinkedIn with 3.7%, the rest of the applications are between 3.7% and 0.2%.

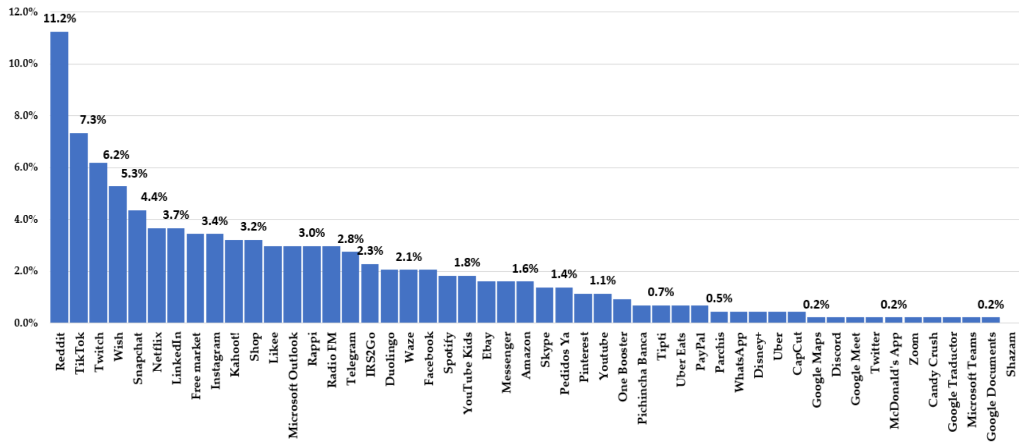

Figure 2. Percentage of accessibility errors of the tools evaluated.

Table 3. Evaluation of mobile applications with the Accessibility Scanner.

Tool	E	Tt	Ce	Tc	IL	EtL	Eet	Id	Ic	TNs
Shazam	0									
Candy Crush	1				1					
Google Traductor	1	1								
Google Documents	1	1								
Twitter	1				1					
McDonald's App	1				1					
Discord	1				1					
Microsoft Teams	1					1				
Google Meet	1	1								
Google Maps	1	1								
Zoom	1	1								
CapCut	2	1			1					
Disney+	2								2	
WhatsApp	2	2								
Parchis	2				2					
Uber	2	1		1						
Pichincha Banca	3	1					2			
Tipti	3	1		1	1					
Uber Eats	3	2						1		
PayPal	3	2		1						
One Booster	4	2			2					
YouTube	5	4						1		
Pinterest	5	4						1		
Skype	6	5		1						
Pedidos Ya	6	2			4					

Table 3. *Cont.*

Tool	E	Tt	Ce	Tc	IL	EtL	Eet	Id	Ic	TNs
Messenger	7	5		1				1		
eBay	7	6				1				
Amazon	7	4		1		2				
YouTube Kids	8	6				1		1		
Spotify	8	8								
Duolingo	9	3		4	1				1	
Waze	9	2			6		1			
Facebook	9	6		1				2		
IRS2Go	10	5			5					
Telegram	12				12					
Rappi	13	1			12					
Microsoft Outlook	13	9				2		2		
Radio FM	13	10			3					
Likee	13	7			6					
Shop	14	5			9					
Kahoot!	14	2			11			1		
Instagram	15	14								1
Free Market	15	1		8	6					
Netflix	16	7		3				4	2	
LinkedIn	16	8		2				1		5
Snapchat	19	3			8	6		2		
Wish	23	2		15	2				3	1
Twitch	27	14			10			3		
TikTok	32	13	7		12					
Reddit	49	34		1	6			8		

Where Total number of Errors = E, Errors Touch target = Tt, Errors Clickable elements = Ce, Errors Text contrast = Tc, Errors Item label = IL, Errors Element type label = EtL, Errors Editable element tag = Eet, Errors Item descriptions = Id, Errors Image contrast = Ic, Errors Type of element not supported = TNs.

This study revealed that even though users highly rate the applications, the applications are not accessible. Table 3 shows the evaluation of accessibility in mobile applications.

Table 4 presents the classification, mobile applications, and the number of accessibility problems. With the data in Table 4, we found that the Pearson's parametric correlation between the ranking of mobile applications and accessibility barriers is positive but weak, with r = 0.245 and p = 0.086. To calculate the correlation, we used IBM SPSS Statistics version 25 software.

Table 4. App ranking vs. accessibility issues.

Ranking	App	Accessibility Issues
4.8	Free Market	15
	Shop	14
4.7	Ebay	7
	One Booster	4
4.6	Amazon	7

Table 4. *Cont.*

Ranking	App	Accessibility Issues
	Candy Crush	1
	Duolingo	9
	Kahoot!	14
	McDonald's App	1
	Microsoft Outlook	13
	Pinterest	5
	Radio FM	13
	Wish	23
4.5	CapCut	2
	Discord	1
	Google Translator	1
	Pichincha Banca	3
	Reddit	49
	Spotify	8
	Telegram	12
	TikTok	32
4.4	Instagram	15
	LinkedIn	16
	Netflix	16
	Shazam	0
	Twitch	27
	Uber	2
	Uber Eats	3
	Waze	9
4.3	Disney+	2
	Google Documents	1
	Likee	13
	Parchis	2
	Rappi	13
	Skype	6
	Snapchat	19
	YouTube Kids	8
4.2	Google Maps	1
	Messenger	7
	PayPal	3
	Twitter	1
	WhatsApp	2
4.1	Facebook	9
	Microsoft Teams	1
	Youtube	5
3.9	IRS2Go	10

Table 4. *Cont.*

Ranking	App	Accessibility Issues
	Zoom	1
3.7	Google Meet	1
3.4	Tipti	3
3.1	Pedidos Ya	6

Figure 3 presents the number of barriers related to the four accessibility principles. The highest number of barriers found in the evaluation of mobile applications corresponds to the operable principle with 53.9%. This scenario implies that a mobile application must ensure all functionality related to navigation. It consists of giving a prudent time to read and understand the content. The times in which the information changes must be sufficiently assimilated so that all potential users can understand them regardless of their condition or age.

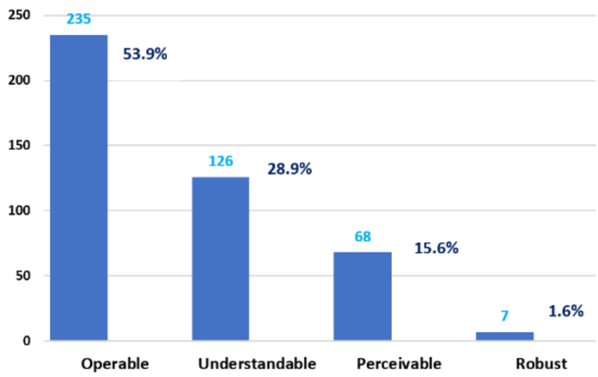

Figure 3. Accessibility principles vs. barriers.

It is essential that the content design does not provoke seizures or epileptic convulsions, so it is necessary to provide means for uncluttered navigation; sometimes, less is more. The principle of understandable with 28.9%; for the mobile application to overcome this principle, it must be easy to read and comprehend. It refers to both the form and the content of an application's texts. They must have a font that all users can read. Furthermore, the application must be predictable; it is related to the operation of the application so that users do not waste time trying to guess what one option or another is for; everything must be clear for a better browsing experience.

The results reveal that the perceivable principle corresponds to 15.6% of all users' contents, and an application interface can be perceived. The interface, images, buttons, and other elements must be accessible, perfectly identifiable and executable by any person in any situation, device, and operating system. Color is also part of the group of relevant indicators since a percentage of the population with some condition cannot perceive them correctly. Robust with 1.6% refers to the application's compatibility with assistive technologies or digital ramps.

5. Discussion

The study found that the most significant number of applications violates the operable principle with 53.9%, followed by understandable with 28.9%. The most frequently repeated problem is the touch target, related to the touch targets, which must have at least 44 by 44 pixels [26].

Most of the application problems related to contrast are related to the perceptible principle with 15.6%, where visually impaired people are involved, 7:1 is for large text, 4.5:1 is for standard text [26]. Less incidence of errors was detected for the robust principle with 1.6%.

The most frequent barriers have a high incidence on the tactile target and the selectable elements, both being a barrier to selecting objects on the screen and impairing the fluidity of navigation.

The research [27] indicates that the social network TikTok is one of the most used media by the young generation; in this study, we found that TikTok has accessibility barriers that make it difficult for all people, including young people, to navigate the application; the most frequent error is related to touch target.

The study [11] indicates that accessible mobile applications are a significant challenge for accessibility experts and suggests verifying compliance with the accessibility guidelines proposed by WCAG 2.1 throughout the application development process. Compared to the results of this study, it is suggested to apply a continuous evaluation process to reduce possible accessibility barriers, using a manual and automatic method as suggested by the study [12].

The results of this study reveal that to achieve adequate accessibility in mobile applications, it is essential to apply WCAG 2.1 according to studies [22,28,29]. In order to (1) improve text contrast, the text and the background must not share the same chromatic range; (2) correct legibility by matching the text with a light contrast ratio of at least 4.5:1 concerning the background; (3) apply an appropriate simple Sans Serif typeface; (4) use headings and descriptions to make known what users cannot otherwise understand or see; (5) clearly define visual content; (6) provide options to customize the application to facilitate navigation; (7) involve a simple design process by eliminating any superfluous information.

Finally, one of the limitations of this study is that an automatic review method was used; although these tools facilitate the automatic review of certain aspects, human evaluation is still necessary. Most tools for accessibility evaluation assess whether the mobile application screens satisfy the WCAG, so in future work, we will apply a manual review method based on heuristics to improve accessibility levels in mobile applications.

6. Conclusions and Future Work

We determined that the most frequent problems are tactile elements and clicks related to the principle of operability. We also identified that 53.21% of apps intended for the "all users" category defined by Google Play lack an adequate level of accessibility.

Among the least accessible categories, social networks represent 28.67%, with Facebook, Instagram, Twitter, and Tiktok. Among the most accessible is the Travel category with the Google Maps application. The "teen users" term defined in the content classification section of Google Play does not identify app issues because they are more familiar with the type of technology.

During the designing of apps, it is essential to test their accessibility at each stage using the Accessibility Scanner. In addition, it is crucial to define the correct contrast between background and text to be within the most appropriate accessibility ranges. We suggest testing the contrast between background and text [30].

Basing accessibility analysis exclusively on an automatic tool is very limited since it neither detects all errors nor are the errors they detect accurate. However, we suggest (1) complementing the automatic review evaluations with a manual method based on heuristics to ensure an adequate level of accessibility in mobile apps; (2) use this study as a starting point to create a software tool using WCAG 2.1 based on artificial intelligence algorithms to help developers evaluate accessibility in mobile apps; and (3) work on country-specific accessibility legislation and solutions that incorporate user feedback to improve accessibility levels with systematization and monitoring tools. Another limitation is that mobile applications available for Ecuador in the Google Play stores were evaluated due to the need for geolocation of device use.

As future work, we suggest (1) including paid mobile apps in future accessibility evaluations; (2) testing accessibility evaluation with users with different types of disabilities; (3) designing automatic review apps that include artificial intelligence algorithms so that people with disabilities can significantly improve their quality of life; (4) evaluating all the mobile applications' screens evaluated in this research to test and compare the influence on the results.

Author Contributions: Conceptualization, P.A.-V.; methodology, P.A.-V.; investigation, P.A.-V., S.Z.-E., and F.M.-V.; writing—original draft preparation, P.A.-V., S.L.-M., S.Z.-E., and F.M.-V.; writing—review and editing, P.A.-V., S.L.-M., S.Z.-E., F.M.-V., S.N.-V., and C.C.; supervision, P.A.-V., S.L.-M.; project administration, P.A.-V. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Universidad de Las Américas-Ecuador, an internal research project INI.PAV.20.01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mayordomo-Martínez, D.; Sánchez-Aarnoutse, J.-C.; Carrillo-De-Gea, J.M.; García-Berná, J.A.; Fernández-Alemán, J.L.; García-Mateos, G. Design and Development of a Mobile App for Accessible Beach Tourism Information for People with Disabilities. *Int. J. Environ. Res. Public Heal.* **2019**, *16*, 2131. [CrossRef] [PubMed]
2. World Health Organization (WHO). World Report on Disability 2011. Available online: https://apps.who.int/iris/rest/bitstreams/53067/retrieve (accessed on 20 July 2021).
3. World Bank Group. World Report on Disability: Main Report. Available online: http://documents.worldbank.org/curated/en/665131468331271288/Main-report (accessed on 20 July 2021).
4. United Nations (UN). Sustainable Development Goals. Available online: https://www.un.org/sustainabledevelopment/ (accessed on 20 July 2021).
5. Statista Research Department. Google Play: Number of Available Apps 2009–2020. Available online: https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/ (accessed on 20 July 2021).
6. World Wide Web Consortium. Web Content Accessibility Guidelines (WCAG) 2.1. Available online: https://www.w3.org/TR/WCAG21/ (accessed on 20 July 2021).
7. World Wide Web Consortium. Mobile Accessibility: How WCAG 2.0 and Other W3C/WAI Guidelines Apply to Mobile. Available online: https://www.w3.org/TR/mobile-accessibility-mapping/ (accessed on 20 July 2021).
8. Google Play. Top Free Apps. Available online: https://play.google.com/store/apps/top (accessed on 20 July 2021).
9. Tomažič, T.; Mišič, K.U. Parliament-citizen Communication in Terms of Local Self-Government and Their Use of Social Media in the European Union. Available online: http://pub.lex-localis.info/index.php/LexLocalis/article/view/1220 (accessed on 20 July 2021).
10. Statista Research Department. Number of Apps Available in Leading App Stores. 2021. Available online: https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/ (accessed on 20 July 2021).
11. Acosta-Vargas, P.; Serrano-Costales, L.; Salvador-Ullauri, L.; Nunes, I.L.; Gonzalez, M.; Acosta-Vargas, P. Toward Accessible Mobile Application Development for Users with Low Vision. In Proceedings of the AHFE 2020 Virtual Conference on Human Factors and Systems Interaction, San Diego, CA, USA, 16–20 July 2020. [CrossRef]
12. Acosta-Vargas, P.; Salvador-Acosta, B.; Salvador-Ullauri, L.; Villegas-Ch., W.; Gonzalez, M. Accessibility in Native Mobile Applications for Users with Disabilities: A Scoping Review. *Appl. Sci.* **2021**, *11*, 5707. [CrossRef]
13. Zhang, X.; Ross, A.S.; Fogarty, J. Robust Annotation of Mobile Application Interfaces in Methods for Accessibility Repair and Enhancement. In Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology, Berlin, Germany, 14 October 2018. [CrossRef]
14. Acosta-Vargas, P.; Zalakeviciute, R.; Luján-Mora, S.; Hernandez, W. Accessibility Evaluation of Mobile Applications for Monitoring Air Quality. Proceedings of ICITS 2019, Quito, Ecuador, 6–8 February 2019. [CrossRef]
15. Yan, S.; Ramachandran, P.G. The Current Status of Accessibility in Mobile Apps. *ACM Trans. Access. Comput.* **2019**, *12*, 1–31. [CrossRef]
16. Ross, A.S.; Zhang, X.; Fogarty, J.; Wobbrock, J.O. An Epidemiology-inspired Large-scale Analysis of Android App Accessibility. *ACM Trans. Access. Comput.* **2020**, *13*, 1–36. [CrossRef]
17. Vendome, C.; Solano, D.; Linan, S.; Linares-Vasquez, M. Can Everyone use my app? An Empirical Study on Accessibility in Android Apps. Proccedings of the 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), Cleveland, OH, USA, 29 September–4 October 2019. [CrossRef]

18. Zhang, X.; de Greef, L.; Swearngin, A.; White, S.; Murray, K.; Yu, L.; Shan, Q.; Nichols, J.; Wu, J.; Fleizach, C.; et al. Screen Recognition: Creating Accessibility Metadata for Mobile Applications from Pixels. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, Yokohama, Japan, 8–13 May 2021. [CrossRef]
19. Gonçalves, T.S.; Ivars-Nicolás, B.; Martínez-Cano, F.J. Mobile Applications Accessibility: An Evaluation of the Local Portuguese Press. *Informatics* **2021**, *8*, 52. [CrossRef]
20. Mateus, D.A.; Silva, C.A.; Eler, M.M.; Freire, A.P. Accessibility of mobile applications. In Proceedings of the 19th Brazilian Symposium on Human Factors in Computing Systems, Diamantina, Brazil, 26–30 October 2020. [CrossRef]
21. Acosta-Vargas, P.; Salvador-Ullauri, L.; Guerrero, J.L.J.; Guevara, C.; Sanchez-Gordon, S.; Calle-Jimenez, T.; Lara-Alvarez, P.; Medina, A.; Nunes, I.L. Accessibility Assessment in Mobile Applications for Android. In Proceedings of the AHFE 2019 International Conference on Human Factors and Systems Interaction, Washington, DC, USA, 24–28 July 2019. [CrossRef]
22. Acosta-Vargas, P.; Salvador-Acosta, B.; Zalakeviciute, R.; Alexandrino, K.; Pérez-Medina, J.-L.; Rybarczyk, Y.; Gonzalez, M. Accessibility Assessment of Mobile Meteorological Applications for Users with Low Vision. In Proceedings of the AHFE 2020 Virtual Conference on Human Factors and Systems Interaction, San Diego, CA, USA, 16–20 July 2020. [CrossRef]
23. Acosta-Vargas, P.; Salvador-Ullauri, L.; Pérez-Medina, J.L.; Zalakeviciute, R.; Hernandez, W. Heuristic Method of Evaluating Accessibility of Mobile in Selected Applications for Air Quality Monitoring. In Proceedings of the AHFE 2019 International Conference on Human Factors and Systems Interaction, Washington, DC, USA, 24–28 July 2019. [CrossRef]
24. Brajnik, G. Measuring Web Accessibility by Estimating Severity of Barriers. Proccedings of the International Conference on Web Information Systems Engineering 2008, Auckland, New Zealand, 1–4 September 2008. [CrossRef]
25. Mantilla-Vaca, F.; Zarate-Estrella, S.; Acosta-Vargas, P. (Dataset) Accessibility Assessment in Mobile Applications for Android. *Mendeley Data* **2021**. [CrossRef]
26. World Wide Web Consortium (W3C) (2021) Understanding WCAG 2.1. Available online: https://www.w3.org/WAI/WCAG21/Understanding/ (accessed on 19 July 2021).
27. Vázquez-Herrero, J.; Negreira-Rey, M.-C.; Rodríguez-Vázquez, A.-I. Intersections between TikTok and TV: Channels and Programmes Thinking Outside the Box. *J. Media* **2021**, *2*, 1. [CrossRef]
28. Carvalho, L.P.; Freire, A.P. Native or web-hybrid apps? An analysis of the adequacy for accessibility of android interface components used with screen readers. In Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems, Joinville, Brazil, 23–27 October 2017. [CrossRef]
29. Salvador-Ullauri, L.; Acosta-Vargas, P.; Gonzalez, M.; Luján-Mora, S. Combined Method for Evaluating Accessibility in Serious Games. *Appl. Sci.* **2020**, *10*, 6324. [CrossRef]
30. Color Contrast Checker. Available online: https://siteimprove.com/es-es/accessibility/color-contrast-checker/ (accessed on 20 July 2021).