



PRÁCTICAS DE ACCESIBILIDAD EN EL DESARROLLO DE SOFTWARE: MAPEO SISTEMÁTICO DE LA LITERATURA *

María Elena Fernández-Castillo**
Juan Carlos Pérez-Arriaga***
María Karen Cortés-Verdín****
Jorge Octavio Ocharán-Hernández*****

Received: 01/11/2023 • Accepted: 07/02/2024

<https://doi.org/10.22395/rium.v23n44a4>

Resumen

Incorporar la accesibilidad en el desarrollo de *software* conlleva beneficios significativos como la inclusión de personas con discapacidad. Reconocer la diversidad de usuarios con diferentes requisitos y necesidades dentro de la creación de software facilita su utilización por la mayor cantidad de usuarios sin importar sus condiciones individuales. El objetivo de esta investigación es obtener una base de conocimientos que permita identificar aquellas prácticas de las que se tenga evidencia de su uso en el desarrollo de software accesible. Se llevó a cabo un Mapeo Sistemático de la Literatura (MSL) enfocado en identificar prácticas que incorporan actividades de accesibilidad en el ciclo de vida del software, resultando en 29 estudios que se ajustaron a los criterios de selección. Los resultados muestran que la fase con más evidencia de incorporación de actividades de accesibilidad en el desarrollo de software es la fase de pruebas, donde se utilizan evaluadores para verificar la accesibilidad dentro de páginas web, seguido de la fase de construcción, donde se utilizan metodologías que incluyen la accesibilidad dentro del proceso de desarrollo de software; en la fase de diseño es común que se incorporen artefactos que guíen el

* El artículo corresponde a la investigación realizada como parte del proyecto “Hacia un modelo de campus accesible: Facultad de Estadística e Informática” del Cuerpo Académico Ingeniería y Tecnologías de Software.

** Estudiante de la licenciatura Ingeniería de Software, Universidad Veracruzana, Región Xalapa – Veracruz (E-mail: mariaelenafdez1@gmail.com) Orcid: <https://orcid.org/0009-0006-6282-6993>

*** Profesor de tiempo completo de la Facultad de Estadística e Informática, Xalapa – Veracruz (E-mail: juaperez@uv.mx) Orcid: <https://orcid.org/0000-0003-2354-2462>

**** Profesora de tiempo completo de la Facultad de Estadística e Informática, Xalapa – Veracruz (E-mail: kcortes@uv.mx) Orcid: <https://orcid.org/0000-0002-6453-180X>

***** Profesor de tiempo completo de la Facultad de Estadística e Informática, Xalapa – Veracruz (E-mail: jocharan@uv.mx) Orcid: <https://orcid.org/0000-0002-2598-1445>

desarrollo de accesible y, por último, en la fase de requisitos es usual emplear la descripción de requisitos para obtener y documentar requisitos de accesibilidad.

Palabras clave: desarrollo de software, ciclo de vida del software, ingeniería de software, software accesible, accesibilidad, discapacidad, técnicas, procesos, lineamientos, mapeo sistemático.

ACCESSIBILITY PRACTICES IN SOFTWARE DEVELOPMENT: SYSTEMATIC LITERATURE MAPPING

Abstract

Incorporating accessibility in software development brings significant benefits such as the inclusion of people with disabilities. Recognizing the diversity of users with different requirements and needs within software development facilitates its use by the greatest number of users regardless of their individual conditions. The objective of this research is to obtain a knowledge base to identify those practices for which there is evidence of their use in the development of accessible software. A Systematic Literature Mapping (SLM) focused on identifying practices that incorporate accessibility activities in the software life cycle was conducted, resulting in 29 studies that met the selection criteria. The results show that the phase with more evidence of incorporation of accessibility activities in software development is the testing phase, when testers are used to verify accessibility within web pages, followed by the construction phase, when methodologies that include accessibility within the software development process are used; in the design phase it is common to incorporate artifacts that guide the development of accessible software and, finally, in the requirements phase it is common to use the requirements description to obtain and document accessibility requirements.

Keywords: software development, software life cycle, software engineering, accessible software, accessibility, disability, techniques, processes, guidelines, systematic mapping.

INTRODUCCIÓN

La accesibilidad es la facilidad para que un grupo de personas con discapacidad puedan utilizar un producto, sistema o entorno para alcanzar un objetivo específico en un contexto de uso concreto [1].

Las personas con discapacidad son aquellas que tienen deficiencias físicas, mentales, intelectuales o sensoriales a largo plazo. Estas interactúan con diversas barreras que impiden su participación plena y efectiva en la sociedad y en igualdad de condiciones con las demás personas [2].

En México, el INEGI define a las personas con discapacidad como aquellas que tienen dificultad para llevar a cabo actividades consideradas básicas, como: ver, escuchar, caminar, recordar o concentrarse, realizar cuidado personal y comunicarse [3].

Frecuentemente el software es desarrollado pensando en usuarios con características similares. Sin embargo, es importante recordar que existen usuarios con requisitos diversos. Una de las ventajas de desarrollar software accesible y compatible con tecnologías de asistencia es que puede ser utilizado por el mayor número de usuarios posibles [4].

Identificar actividades clave para incorporar accesibilidad en el desarrollo de software facilita su integración por distintos equipos de desarrollo en sus procesos. El propósito de este MSL es recopilar prácticas de accesibilidad en el desarrollo de software que tengan evidencia de su aplicación en las diferentes fases del proceso.

El presente MSL está estructurado como se describe a continuación: la Introducción abordó los antecedentes y trabajos relacionados. La sección 1 detalla el método de investigación empleado, mientras que la sección 2 presenta los resultados de la investigación. Finalmente, la sección 3 describe las reflexiones y conclusiones derivadas de este estudio.

1. MÉTODO DE INVESTIGACIÓN

Para la conducción de esta investigación se siguió el método propuesto por Kitchenham *et al.* [5], el cual se describe a continuación.

1.1. Preguntas de investigación

Para esta investigación se definieron tres preguntas de investigación que se muestran en la tabla 1.

Tabla 1. Preguntas de investigación

Pregunta de investigación	Motivación
RQ1. ¿Cuáles son las herramientas, artefactos o actividades orientadas al desarrollo de software accesible?	Conocer cuáles son las herramientas, artefactos o actividades que promueven la accesibilidad en el desarrollo de software.
RQ2. ¿Cuáles son las fases en las que se consideran aspectos de accesibilidad en el desarrollo de software?	Determinar en qué fase del desarrollo se presta más atención a los aspectos de accesibilidad.
RQ3. ¿Cuáles son las herramientas, actividades o artefactos de los que se tiene evidencia de su uso en el desarrollo de software accesible?	Identificar las herramientas, actividades o artefactos que se han utilizado en alguna de las fases para el desarrollo de software accesible.

Fuente: elaboración propia.

1.2. Proceso de búsqueda

Se realizó un procedimiento de búsqueda definiendo un conjunto de términos que conformaron la cadena de búsqueda principal. Esta cadena se aplicó en las bases de datos seleccionadas y se evaluó su validez utilizando el estándar Quasi-Gold propuesto por Zhang *et al.* [6]. La cadena principal se introdujo en la base de datos de IEEE Xplore y logró un índice de recuperación superior al 80 %. Los incisos a, b y c detallan el procedimiento del proceso de búsqueda.

Cadena de búsqueda: (“*accessibility*” OR “*inclusive*” OR “*disabilities*”) AND (“*software engineering*” OR “*software development*” OR “*software life cycle*”) AND (“*techniques*” OR “*method*” OR “*process*” OR “*model*” OR “*guidelines*”)

a. Selección de fuentes

Se consideraron las áreas de ingeniería de software y ciencias de la computación para la selección de las siguientes fuentes de búsqueda: IEEE Xplore, ScienceDirect, Springer Link y ACM Digital Library.

b. Selección de estudios primarios

Los términos de inclusión y exclusión empleados para determinar la selección de un estudio primario se describen a continuación.

- *Criterios de inclusión*: estudios publicados entre 2017 y 2022, escritos en inglés y que a partir de la lectura del título o resumen respondan a alguna pregunta de investigación.
- *Criterios de exclusión*: estudios a los que no se tiene el acceso completo y que no sean artículos de investigación o estudios repetidos o duplicados.
- Procedimiento de selección de estudios primarios.

La cadena de búsqueda elaborada se introdujo en cada una de las fuentes seleccionadas; además, se aplicaron los criterios de exclusión e inclusión. El proceso se detalla en la tabla 2.

Tabla 2. Proceso de selección de estudios primarios

Resultados al aplicar la cadena de búsqueda			
ACM	IEEE	Science Direct	Springer Link
813	336	19.646	29.993
CI1: estudios publicados entre 2017 y 2022; CE1: artículos a los que no se tiene el acceso completo.			
ACM	IEEE	Science Direct	Springer Link
415	170	3.627	1.502
CI2: artículos escritos en inglés; CE2: artículos que no son de investigación.			
ACM	IEEE	Science Direct	Springer Link
293	170	929	753
CI3: artículos que a partir de la lectura del título o resumen respondan a alguna pregunta de investigación; CE3: estudios repetidos o duplicados.			
ACM	IEEE	Science Direct	Springer Link
0	23	2	4

Fuente: elaboración propia.

c. Extracción y síntesis

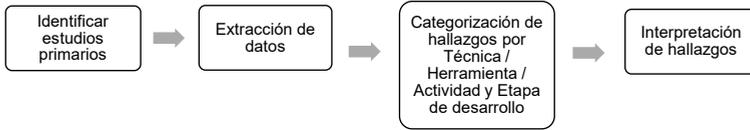
El MSL se realizó en el primer semestre del año 2023 y la investigación fue adelantada por dos investigadores.

La extracción de datos fue realizada con base en las preguntas de investigación, identificando 29 artículos que las contestan y categorizando estos hallazgos por herramienta, artefacto y actividad, así como fase de desarrollo; por último, se documentó la evidencia que se tenía del uso de estas herramientas, artefactos y actividades en el proceso de desarrollo de software accesible.

Se siguió la metodología propuesta por Popay *et al.* [7] para llevar a cabo una síntesis narrativa, que se utiliza principalmente en revisiones sistemáticas para evaluar la efectividad de intervenciones o factores relacionados con su implementación en el campo de las ciencias de la salud. La síntesis narrativa de este trabajo se realizó siguiendo esta metodología, ya que es aplicable tanto a investigaciones cualitativas como cuantitativas.

El proceso de síntesis se muestra en la figura 1. El proceso completo, así como las referencias de los estudios primarios analizados, se encuentran en: Proceso Síntesis.xlsx.

Figura 1. Proceso de síntesis empleado en la investigación

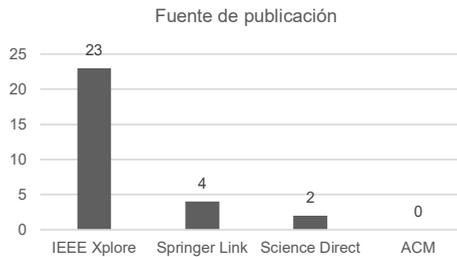


Fuente: diagrama basado en la metodología de síntesis narrativa propuesto por Popay *et al.* [7].

2. RESULTADOS

Se seleccionaron un total de 29 estudios primarios y la mayoría de ellos fueron obtenidos de IEEE Xplore con un total de 23 artículos, seguido de Springer Link con cuatro artículos y Science Direct con dos artículos. La figura 2 muestra la distribución de los estudios primarios encontrados por fuente de publicación y la figura 3 muestra la distribución de los estudios por año de publicación.

Figura 2. Distribución de estudios por fuente de publicación



Fuente: elaboración propia.

Figura 3. Distribución de estudios por año de publicación.



Fuente: elaboración propia.

RQ1: ¿Cuáles son las herramientas, artefactos o actividades orientadas al desarrollo de software accesible?

En el análisis de los estudios se encontraron 22 herramientas utilizadas para integrar la accesibilidad en el desarrollo. La herramienta más utilizada es AChecker, que permite evaluar el contenido HTML en busca de problemas de accesibilidad. Los estudios S9, S11 y S13 la emplearon para la evaluación de accesibilidad a 10 páginas web de bancos y 20 sitios de eCommerce de Pakistán, así como para la evaluación del sitio web de un museo en Perú. La segunda herramienta más empleada es WAVE Tool, que se utiliza para evaluar la accesibilidad en páginas web. Esta herramienta clasifica los errores en tres colores: rojo para los que deben corregirse, amarillo para advertir sobre elementos que podrían causar problemas de accesibilidad y verde para los elementos implementados correctamente. Tres estudios destacados son S17, S19 y S25, los cuales emplearon esta herramienta para evaluar la accesibilidad de 51 universidades globales, 10 sitios web bancarios y 76 sitios web de salud en Chipre. TAW Tool es la tercera herramienta más utilizada, con tres artículos. Se trata de herramientas que evalúan la accesibilidad en un sitio web con base en los lineamientos de la WCAG 2.1. Los estudios S10 y S11 la utilizaron para la evaluación de 10 sitios web de bancos y 20 sitios de eCommerce de Pakistán y el estudio S17 para el análisis de 45 sitios web de universidades de Bosnia y Herzegovina. La tabla 3 muestra la lista de herramientas que conllevan al desarrollo de software accesible. La figura 4 muestra la frecuencia en la que aparecieron las 22 herramientas encontradas en los estudios primarios analizados.

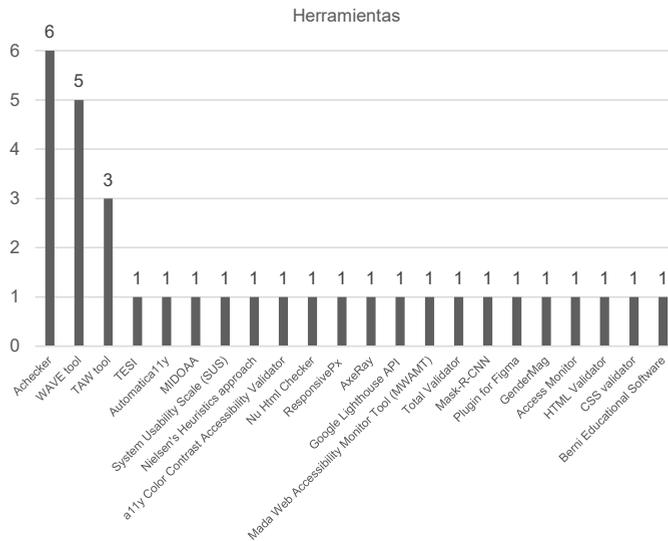
Tabla 3. Herramientas orientadas al desarrollo de software accesible

Herramienta	Artículo(s) (número en referencias)	Objetivo
AChecker	S9, S11, S13, S17, S25 y S26.	Evaluar la accesibilidad analizando contenidos HTML y CSS
WAVE Tool	S7, S9, S13, S15 y S26.	Evaluar la accesibilidad con base en los lineamientos de la WCAG 2.1
TAW Tool	S9, S11 y S17.	Evaluar la accesibilidad con base en los lineamientos de la WCAG 2.1
TESI	S2.	Mejorar el proceso de integración social de los alumnos con problemas de comunicación verbal mediante el uso de software personalizado
Automatica11y	S3.	Mejorar el código fuente considerando aspectos de accesibilidad ingresado a la herramienta y genera un informe de los resultados
MIDOAA	S4.	Desarrollar objetos de aprendizaje
Escala SUS (System Usability Scale)	S11.	Evaluar la usabilidad desde la perspectiva del usuario
Cuestionario de usabilidad basado en las heurísticas de Jakob Nielsen	S11.	Evaluar la usabilidad desde la perspectiva de los expertos
a11y Color Contrast Accessibility Validator	S13.	Analizar el contraste de color con base en los lineamientos WCAG 2.1
Nu Html Checker	S13.	Validar código en documentos HTML
ResponsivePx	S13.	Validar si una página web es responsiva

Herramienta	Artículo(s) (número en referencias)	Objetivo
AxeRay	S14.	Automatizar las pruebas de accesibilidad web desde una perspectiva semántica
Google Lighthouse API	S16	Evaluar páginas web y generar un informe de la evaluación
Mada Web Accessibility Monitor Tool (MWAMT)	S16	Identificar y priorizar los resultados de las pruebas de accesibilidad dentro del desarrollo ágil
Total Validator Pro	S17.	Validar HTML, CSS, DOM conforme con los lineamientos WCAG 2.1, US Section 508 y estándar ARIA
Mask-R-CNN	S18.	Detectar objetos
Plugin for Figma	S21.	Evaluar la accesibilidad en prototipos creados en Figma
GenderMag	S23.	Evaluar la accesibilidad utilizando personas
Access Monitor	S25.	Evaluar la accesibilidad con base en los lineamientos WCAG 2.1 ingresando el enlace o código HTML de una página web
HTML Validator	S26.	Comprobar la validez de documentos web en HTML, XHTML, SMIL, etcétera.
CSS Validator	S26.	Validar la estructura de documentos CSS
Berni Educational Software	S29.	Mejorar las dificultades de niños con dislexia

Fuente: elaboración propia.

Figura 4. Frecuencia de herramientas en estudios primarios



Fuente: elaboración propia.

En el análisis de los estudios se encontraron 21 artefactos utilizados en el desarrollo de software accesible. Los prototipos son el artefacto del que se encontró más evidencia de su uso con un total de 6 artículos. Usualmente se utilizan para proporcionar una

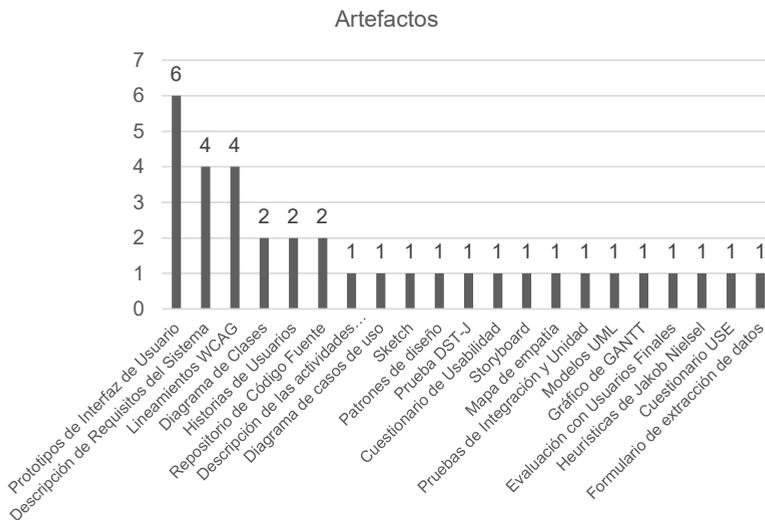
versión preliminar del sistema. El artículo S1 utilizó este artefacto para el desarrollo de un modelo para niños con dislexia, el artículo S5 para el desarrollo de una herramienta web que apoya en la terapia a niños con labio leporino o paladar hendido y el artículo S28 para el desarrollo de un videojuego educativo para niños con discapacidad auditiva. El uso de estándares como las pautas de la WCAG representan el segundo artefacto más utilizado con un total de cuatro artículos; principalmente se utilizó en estudios como el S12 para la evaluación del modelo de software Shift Left AI1y. La especificación de requisitos constituye el tercer artefacto más utilizado con un total de cuatro artículos. Este documento contiene los requisitos del sistema, documenta requisitos funcionales, atributos de calidad, restricciones y más. El artículo S5 lo utilizó en el desarrollo de una herramienta para el apoyo en la terapia a niños con labio leporino o paladar hendido y el artículo S8 para la elaboración de un modelo para el desarrollo de juegos educativos. La tabla 4 enlista los artefactos orientados al desarrollo accesible. La figura 5 muestra la frecuencia en la que aparecieron las herramientas en los estudios primarios.

Tabla 4. Artefactos orientados al desarrollo accesible

Artefactos	Estudio(s) primario(s)
Descripciones de las actividades principales de actores	S1.
Diagramas de casos de uso	S1.
Diagrama de clases	S1, S8.
Sketchs	S1.
Prototipos de diseño de interfaz	S1, S2, S5, S12, S20, S28.
Patrones de Diseño	S1.
Prueba DST-J	S1.
Cuestionario de usabilidad	S1.
Descripción de los requerimientos del sistema	S2, S4, S5, S8.
Repositorio de código fuente	S2, S4.
Storyboard	S4.
Mapa de empatía	S4.
Pruebas de integración y unidad	S5.
Historias de usuario	S6, S12.
Modelos UML	S8.
Gráfico de GANTT	S8.
Evaluación con usuarios finales	S8.
Resultados del cuestionario utilizando las heurísticas de Jakob Nielsen	S9.
Resultados del cuestionario USE	S9.
Formulario de extracción de datos	S10.
Lineamientos WCAG	S12, S13, S17, S21.

Fuente: elaboración propia.

Figura 5. Frecuencia de artefactos en estudios primarios



Fuente: elaboración propia.

En el análisis de los 29 artículos se encontraron 9 actividades destacadas en el desarrollo de software accesible. Las actividades más utilizadas son la integración de SCRUM en el proceso de desarrollo de software; artículos como S4 y S28 la utilizaron para el desarrollo de modelos inclusivos para la atención de niños en situación de discapacidad. El enfoque centrado en el usuario se empleó en el artículo S1 para el desarrollo de una aplicación para niños con dislexia y en S19 para la evaluación de una aplicación para usuarios con discapacidad visual.

La documentación de las herramientas, artefactos y actividades utilizadas en el desarrollo de software proporciona una base de conocimientos para que desarrolladores y profesionales en la industria del software puedan consultar las prácticas relacionadas con el desarrollo de software accesible. La tabla 5 enlista las actividades orientadas al desarrollo de software accesible. La figura 6 muestra la frecuencia en la que aparecieron las nueve actividades en los estudios primarios analizados.

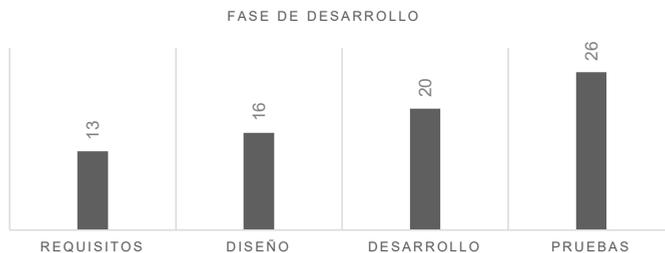
TABLA 5. Actividades orientadas al desarrollo de software accesible

Actividades	Artículo(s)
Uso del enfoque centrado en el usuario para considerar las perspectivas de los interesados desde las primeras etapas	S1, S19, S24.
El uso de la metodología Extreme Programming (XP eXtreme Programming), por su característica iterativa e incremental, permite evaluar los requisitos de accesibilidad de forma rápida	S5, S28.
Añadir las fases de pruebas de accesibilidad, correcciones de accesibilidad y revisiones de accesibilidad a la metodología SCRUM	S4, S6, S28.

Actividades	Artículo(s)
Integración de un experto en usabilidad en el proceso Ágil	S7, S24.
Creación del modelo Shift Left a11y, que integra la accesibilidad desde el inicio del desarrollo de software incluyendo requerimientos de accesibilidad y evaluación del cumplimiento de esos requisitos en la etapa de pruebas	S12.
Entrenamiento del modelo CNN para la pronunciación de seis colores para el desarrollo de una aplicación que ayude a niños con dislexia	S22.
Utilización del enfoque systematic inclusivity fault localization para la localización de errores de inclusividad	S23.
Realización del modelo RiD que integra la accesibilidad en el ciclo de vida del software	S27.
Evaluación de usabilidad mediante un método de evaluación heurística realizado con expertos en usabilidad	S26.

Fuente: elaboración propia.

Figura 6. Frecuencia de actividades en estudios primarios



Fuente: elaboración propia.

RQ2. ¿Cuáles son las fases en las que se consideran aspectos de accesibilidad en el desarrollo de software?

Es en la fase de pruebas en la que se consideraron más aspectos de accesibilidad, con un total de 26 estudios. En esta fase se utilizan herramientas para la evaluación de accesibilidad en sistemas desarrollados; algunas de las más utilizadas son AChecker, con seis artículos y WAVE Tool con cinco artículos. La segunda fase más considerada es la de desarrollo, con un total de 20 artículos; artículos como el S5 y S28 utilizaron metodologías eXtreme Programming en esta fase. La fase de diseño cuenta con 16 artículos relacionados en los que se destaca el uso de prototipos de interfaz de usuario por parte de los artículos S1, S2, S5, S12, S20, S28, con la finalidad de desarrollar sistemas que puedan ser utilizados por personas en situación de discapacidad. Finalmente, la fase de requisitos cuenta con 13 artículos en los que mayormente se utilizó la descripción de los requisitos del sistema por parte de los estudios S2, S4, S5 y S8. La tabla 6 muestra las fases de desarrollo en las que se consideran aspectos de accesibilidad. La frecuencia de estos aspectos se muestra en la figura 7.

Tabla 6. Fases de desarrollo en las que se consideran aspectos de accesibilidad

Fase	Estudios primarios identificados
Requisitos	S1, S2, S4, S5, S6, S7, S8, S12, S19, S24, S17, S28, S29.
Diseño	S1, S2, S4, S5, S6, S7, S8, S12, S17, S19, S20, S21, S24, S27, S28, S29.
Desarrollo	S1, S2, S4, S5, S6, S7, S8, S9, S12, S13, S15, S17, S18, S19, S21, S22, S24, S27, S28, S29.
Pruebas	S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S19, S21, S23, S24, S25, S26, S27, S28, S29.

Fuente: elaboración propia.

FIGURA 7. Fases de desarrollo en las que se encontró evidencia de aspectos de accesibilidad



Fuente: elaboración propia.

a. Discusión

Esta investigación identificó una gran variedad de herramientas, artefactos y actividades que son utilizados en las fases del ciclo de vida del desarrollo del software. Sin embargo, se destaca la recopilación de la evidencia del uso de estas como un diferenciador de investigaciones similares.

Dentro de las herramientas es común que se empleen evaluadores de accesibilidad web como WAVE Tool y AChecker; en el caso de los artefactos es habitual la realización de prototipos de interfaz de usuario, la utilización de estándares como el WCAG 2.1, así como la elicitación de requisitos de accesibilidad. Dentro de las actividades más frecuentes están el uso, adaptación o creación de modelos, metodologías y *frameworks* como SCRUM y el Diseño Centrado en el Usuario.

Se recomienda aplicar modelos que integren la accesibilidad a través del ciclo de vida del software; en S12 se propone el modelo Shift Left al1y, que introduce una metodología que incluye desde la elicitación de requisitos de accesibilidad hasta pruebas de accesibilidad, haciendo énfasis en que las pruebas automatizadas no sustituyen las pruebas manuales ni la retroalimentación de usuarios reales.

En la fase de requisitos se aconseja el uso de la descripción de requisitos del sistema, un artefacto comúnmente empleado. Para garantizar la correcta aplicación de

los requisitos de accesibilidad, se recomienda el prototipado de la interfaz de usuario. Si el producto está en desarrollo, se sugiere seguir estándares o modelos de accesibilidad que encaminen el desarrollo hacia la accesibilidad. En la etapa de pruebas, se recomiendan evaluadores de accesibilidad, pruebas unitarias, pruebas de integración y pruebas con usuarios reales.

La recopilación de las herramientas, actividades y artefactos encontrados en esta investigación, así como su forma de aplicación, servirá para que los profesionales de desarrollo de software identifiquen las prácticas que se utilizan en la producción de sistemas accesibles, así como las fases en las que se implementan. Adicionalmente, los futuros investigadores podrán utilizar esta información para conocer el estado de la consideración de accesibilidad dentro de la elaboración de software accesible.

Sobre las amenazas a la validez, en esta investigación se identificaron algunas amenazas como la falta de acceso a los documentos completos encontrados en la base de datos ACM Digital Library; por lo tanto, solo se consideraron aquellos a los que se tenía acceso, dejando de lado algunos recursos de paga. Otras posibles amenazas es la posibilidad de que los investigadores que llevaron a cabo las prácticas hayan omitido información relevante, lo que podría conducir a la alteración de los resultados. Para mitigar estas amenazas, se empleó un formato de extracción y se amplió el período de investigación, lo que permitió incluir un mayor número de estudios primarios.

3. CONCLUSIONES

En esta investigación, se identificaron un total de 29 artículos que presentan evidencia de la integración de accesibilidad en el software. Se hallaron un total de nueve actividades, 22 herramientas y 21 artefactos relacionados con este tema. Es relevante destacar que la fase con la mayor cantidad de evidencia de la integración de accesibilidad es la fase de pruebas, con un total de 26 artículos haciendo referencia a esta etapa. Sin embargo, es importante señalar que solo se encontraron 13 artículos que ofrecen evidencia del uso de herramientas, artefactos o actividades en la fase de requisitos. La integración de accesibilidad dentro del desarrollo de software promueve el uso igualitario de las tecnologías de la información y, por lo tanto, la agrupación de la información encontrada por tipo de discapacidad resulta relevante como una investigación futura. Esta investigación pretende servir como documento de referencia que sea de apoyo a los profesionales de software que deseen incorporar accesibilidad en los productos desarrollados.

AGRADECIMIENTOS

Agradecimientos a la Facultad de Estadística e Informática de la Universidad Veracruzana por el apoyo para la realización de la investigación, escritura y publicación de este artículo.

REFERENCIAS

- [1] ISO/IEC. *ISO/IEC Guide 71:2014. Guidelines for standards developers to address the needs of older persons and persons with disabilities*, 2nd rev ed. International Organization for Standardization, 2014. <https://www.iso.org/standard/57385.html>
- [2] Naciones Unidas. *Convención de las Naciones Unidas sobre los Derechos de las Personas con Discapacidad y su Protocolo Facultativo*, 13 de diciembre de 2006. Organización de las Naciones Unidas, 2008. <https://www.un.org/disabilities/documents/convention/convoptprot-s.pdf>
- [3] INEGI. Discapacidad en México. Instituto Nacional de Estadística y Geografía, 2020. <https://cuentame.inegi.org.mx/poblacion/discapacidad.aspx>
- [4] Nganji J, Nggada S. Disability-Aware Software Engineering for Improved System Accessibility and Usability. *International Journal of Software Engineering and Its Applications*, 2011; 5: 47-62.
- [5] Kitchenham B, Budgen D, Brereton P. Evidence-Based Software Engineering and Systematic Reviews. *Journal of Systems and Software*, 2015; 95: 1-14.
- [6] Zhang H, Babar MA, Tell P. Identifying Relevant Studies in Software Engineering. *Information and Software Technology*, 2011 Jun; 53 (6): 625-637. <https://doi.org/10.1016/j.infsof.2010.12.010>
- [7] Popay J, Roberts H, Sowden A, Petticrew M, Arai L, Rodgers M. *et al. Guidance on the Conduct of Narrative Synthesis in Systematic Reviews*. A Product from the ESRC Methods Programme, 2006 Apr.

Referencias de los estudios

- [S1] Arteaga JM, Ifigenia D, Rivera P. A Process Model to Develop Educational Applications for Children with Dyslexia. In: *2018 6th International Conference in Software Engineering Research and Innovation (CONISOFT)*, 2018. <https://doi.org/10.1109/CONISOFT.2018.00019>
- [S2] Albert MJ, Pérez-Molina C, Mudarra MJ, García M, Castro M, Mileva M, Paulov N. The TESI Project: An Adaptative Personalized System for Creating Expression Tools in Social Inclusion of Disadvantage Learners. In: *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*, Marrakech, Marruecos, 2018 Oct 21-27: 396-397. <https://doi.org/10.1109/CiST.2018.8596475>

- [S3] Ikhsan IN, Catur Candra MZ. Automatically: An Automated Refactoring Method and Tool for Improving Web Accessibility. In: *2018 5th International Conference on Data and Software Engineering (ICoDSE)*, Mataram, Indonesia, 2018 Nov 7-8: 1-6. <https://doi.org/10.1109/ICODSE.2018.8705894>
- [S4] Mourão AB, Magalhães Netto JF. MIDOAA: Inclusive Model of Development of Accessible Learning Objects. In: *2018 IEEE Frontiers in Education Conference (FIE)*, San José, CA, 2018 Oct 3-6: 1-8. <https://doi.org/10.1109/FIE.2018.8658494>
- [S5] Estrada-Cota I, Carreño-León MA, Sandoval-Bringas JA, Leyva-Carrillo AA, Gutiérrez-Sandoval AJ. Design of a Web Tool to Support Language Therapy for Children with Cleft Lip and / or Palate. In: *2019 International Conference on Inclusive Technologies and Education (CONTIE)*, San José del Cabo, México, 2019: 164-1643. <https://doi.org/10.1109/CONTIE49246.2019.00039>
- [S6] Romero-Chacón V, Muir-Camacho H, Rodríguez-González J, Gómez-Blanco A, Chacón-Rivas M. Adapting SCRUM Methodology to Develop Accessible Web Sites. In: *2019 International Conference on Inclusive Technologies and Education (CONTIE)*, San José del Cabo, México, 2019: 112-1124. <https://doi.org/10.1109/CONTIE49246.2019.00029>
- [S7] Fathauer L, Rao DM. Accessibility in an educational software system: Experiences and Design Tips. In: *2018 IEEE Frontiers in Education Conference (FIE)*, Covington, KY, USA, 2019: 1-8. <https://doi.org/10.1109/FIE43999.2019.9028402>
- [S8] Avila-Pesantez D, Delgadillo R, Rivera LA. Proposal of a Conceptual Model for Serious Games Design: A Case Study in Children with Learning Disabilities. *IEEE Access*, 2019: 7: 161017-161033, 2019. <https://doi.org/10.1109/ACCESS.2019.2951380>
- [S9] Fatima K, Bawany NZ, Bukhari M. Usability and Accessibility Evaluation of Banking Websites. In: *2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, Depok, Indonesia, 2020 Oct 17-18: 247-256. <https://doi.org/10.1109/ICACSIS51025.2020.9263083>
- [S10] Di-Gregorio M, Di-Nucci D, Palomba F, Vitiello G. The Making of Accessible Android Applications: An Empirical Study on the State of the Practice. In: *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, Adelaide, SA, Australia, 2020: 857-861. <https://doi.org/10.1109/ICSME46990.2020.00112>
- [S11] Hamid S, Bawany NZ, Zahoor K. Assessing Ecommerce Websites: Usability and Accessibility Study. In: *2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, Depok, Indonesia, 2020 Oct 17-18: 199-204. <https://doi.org/10.1109/ICACSIS51025.2020.9263162>
- [S12] Armas L, Rojas H, Rentería R. Proposal for an Accessible Software Development Model. In: *2020 3rd International Conference of Inclusive Technology and Education (CONTIE)*, Baja California Sur, México, 2020 Oct 28-30: 104-109. <https://doi.org/10.1109/CONTIE51334.2020.00028>

- [S13] Rojas H, Rentería R, Acosta E, Arévalo H, Pilares M. Application of Accessibility Guidelines in a Virtual Museum. In: *2020 3rd International Conference of Inclusive Technology and Education (CONTIE)*, Baja California Sur, México, 2020 Oct 28-30: 73-79. <https://doi.org/10.1109/CONTIE51334.2020.00022>
- [S14] Bajammal M, Mesbah A. Semantic Web Accessibility Testing via Hierarchical Visual Analysis. In: *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, Madrid, España, 2021 May 22-30: 1610-1621. <https://doi.org/10.1109/ICSE43902.2021.00143>
- [S15] Iseri EI, Uyar K, İlhan U. The Accessibility of Health Related Websites of Cyprus Island. In: *2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Ankara, Turquía, 2021 Oct 21-23: 581-586. <https://doi.org/10.1109/ISMSIT52890.2021.9604555>
- [S16] Al Jabor AN, Adnan F, Park M, Othman A. Mada Web Accessibility Monitor Tool. In: *2021 8th International Conference on ICT & Accessibility (ICTA)*, Túnez, Túnez, 2021 Dec 8-10: 1-5. <https://doi.org/10.1109/ICTA54582.2021.9809423>
- [S17] Čeke D, Kunosić S. Accessibility Analysis of the Bosnia and Herzegovina Universities' Websites in the Relation to Their Position on the Webometrics Ranking List. In: *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, Opatija, Croacia, 2021: 1624-1629. <https://doi.org/10.23919/MIPRO52101.2021.9596966>
- [S18] Buddhika AS, Wijesekera PK, Hasintha-Kavindu DG, Dinusha DS, Samaratunge-Arachchillage USS, Kuruppu TA. Smart Photo Editor for Differently-abled People using Assistive Technology. In: *2021 6th International Conference on Information Technology Research (ICITR)*, Moratuwa, Sri Lanka, 2021 Dec 1-3: 1-6. <https://doi.org/10.1109/ICITR54349.2021.9657285>
- [S19] Shera MA, Iqbal MW, Naqvi MR, Shahzad SK, Sajjad MH, Safqat AR, Saeed MM. Usability Evaluation of Blind and Visually Impaired Interface in Solving the Accessibility Problems. In: *2021 International Conference on Innovative Computing (ICIC)*, Lahore, Pakistán, 2021 Nov 9-10: 1-6. <https://doi.org/10.1109/ICIC53490.2021.9693084>
- [S20] Liu J, Shi J, Xie J, Zhang X, Zhang Z, Grundy J, Kanij T. A Curated Personas and Design Guidelines Tool for Better Supporting Diverse End-users. In: *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, Los Alamitos, CA, USA, 2022: 1606-1613. <https://doi.org/10.1109/COMPSAC54236.2022.00255>
- [S21] Kashif S, Ahmad M, Shahid M, Habib MA, Rizwan K. Development of An Automated Accessibility Evaluation Plugin Tool for Mobile Applications. In: *2022 3rd International Conference on Innovations in Computer Science & Software Engineering (ICONICS)*, Karachi, Pakistán, 2022 Dec 14-15: 1-10. <https://doi.org/10.1109/ICONICS56716.2022.10100578>
- [S22] Muthumal SADM, Neranga KT, Harshanath SMB, Sandeepa VDRP, Lihinikaduwa DNR, Rajapaksha UUSK. Mobile and Simulation-based Approach to reduce the Dyslexia with children Learning Disabilities. In: *2022 IEEE 10th Region 10 Humanitarian Technology*

- Conference (R10-HTC)*, Hyderabad, India, 2022 Sep 16-18: 311-317. <https://doi.org/10.1109/R10-HTC54060.2022.9929822>
- [S23] Guizani M, Steinmacher I, Emard J, Fallatah A, Burnett M, Sarma A. How to Debug Inclusivity Bugs?: A Debugging Process with Information Architecture. In: *ICSE-SEIS '22: Proceedings of the 2022 IEEE/ACM 44th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, Pittsburgh, PA, USA, 2022 May: 90-101. <https://doi.org/10.1145/3510458.3513009>
- [S24] Sousa e Silva J, Gonçalves R, Branco F, Pereira A, Au-Yong-Oliveira M, Martins J. Accessible software development: a conceptual model proposal. *Universal Access in the Information Society*, 2019 Jul; 18: 703–716. <https://doi.org/10.1007/s10209-019-00688-5>
- [S25] Niom T, Lin F. Accessibility of COVID-19 Websites of Asian Countries: An Evaluation Using Automated Tools. *SN Computer Science*, 2022 Sep; 3 (article number 498). <https://doi.org/10.1007/s42979-022-01412-6>
- [S26] Mohammadi MK, Esichaikul V, Mohammadi A. Accessibility and usability evaluation of university websites in Afghanistan: a comparison between public and private universities. *Universal Access in the Information Society*, 2022 Jul; 23: 955-974. <https://doi.org/10.1007/s10209-022-00896-6>
- [S27] Moreno Martínez C, Recas Piorno J, Escribano Otero JJ, Guijarro Mata-García M. Responsive inclusive design (RiD): a new model for inclusive software development. *Universal Access in the Information Society*, 2022 Jul; 22: 893-902. <https://doi.org/10.1007/s10209-022-00893-9>
- [S28] Zapata Lesmes C, Acosta-Solano J, Blanquicett Benavides L, Umaña Ibáñez SF. Design and Production of Educational Video Games for the Inclusion of Deaf Children. *Procedia Computer Science*, 2022; 198: 626-631. <https://doi.org/10.1016/j.procs.2021.12.297>
- [S29] Romero A, Garay U, Tejada E, López de la Serna A. Efficacy of Berni: A software for preschoolers at risk of dyslexia. *International Journal of Child-Computer Interaction*, 2023 Dec; 38: 100411. <https://doi.org/10.1016/j.ijcci.2021.100411>