



Resultados iniciales de una evaluación cualitativa sobre la adherencia de una estrategia para mejorar la realización y gestión de las pruebas de regresión en pequeñas organizaciones desarrolladoras de software

Initial results of a qualitative evaluation of the adherence of a strategy to improve the implementation and management of regression testing in small software development organizations

Edgar López Cruz

División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, Oaxaca, México
eklc.2324@gs.utm.mx

Iván García Pacheco

División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, Oaxaca, México
ivan@mixteco.utm.mx
ORCID: 0000-0002-7594-6410

Carla Pacheco Agüero

División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, Oaxaca, México
leninca@mixteco.utm.mx
ORCID: 0000-0003-0925-6048



<https://doi.org/10.36825/RITI.13.31.002>

Recibido: Junio 22, 2025

Aceptado: Septiembre 30, 2025

Resumen: Las pruebas de regresión que se ejecutan sobre un producto de software garantizan que las modificaciones realizadas en este no generen nuevos errores que afecten negativamente las funcionalidades ya implementadas. Sin embargo, a menudo, la definición de una estrategia para gestionar este tipo de pruebas se dificulta en las pequeñas organizaciones desarrolladoras de software ya que la mayoría enfrenta limitaciones para disponer de recursos financieros, personal capacitado, herramientas de soporte, y experiencia. Por lo tanto, este estudio presenta una estrategia enfocada a este tipo de organizaciones que se basa en principios de la Mejora de Procesos de Software y la Gestión de Conocimiento. En este sentido, como una primera fase previa a la implementación de dicha estrategia en contextos reales, se muestra el diseño y conducción de una evaluación cualitativa que involucró a 135 profesionistas de la industria de software, quienes determinaron la posible adherencia de esta estrategia al contexto de las pequeñas organizaciones. Los resultados obtenidos permitieron evaluar la estrategia y establecer la pauta para diseñar una herramienta computacional de soporte que facilite su evaluación empírica en este tipo de organizaciones.

Palabras clave: *Pruebas de Regresión, Mejora del Proceso, Activos del Proceso, Pequeñas Organizaciones, Adherencia.*

Abstract: Regression tests running on a software product ensure that any modifications carried out do not generate new errors that negatively affect already implemented functionalities. However, defining a strategy for managing this type of testing is often difficult in small-sized software development organizations, as most face limitations in terms of financial resources, trained personnel, support tools, and experience. Therefore, this study introduces a strategy focused on these types of organizations based on the principles of Software Process Improvement and Knowledge Management. In this regard, as a first phase prior to implementing this strategy in real-life contexts, the design and implementation of a qualitative evaluation involving 135 software industry professionals is presented. These professionals determined the potential suitability of this strategy in the context of small organizations. The results obtained allowed for an evaluation of the strategy and established guidelines for designing a supporting computational tool to facilitate its empirical evaluation in these types of organizations.

Keywords: *Regression Testing, Process Improvement, Process Assets, Small Organizations, Adherence.*

1. Introducción

Las pruebas de regresión representan una estrategia fundamental de éxito para el proceso de aseguramiento de la calidad del software, ya que permiten asegurar la calidad del producto mediante la repetición de pruebas previas en versiones actualizadas del software [1]. De acuerdo con [2], este tipo de pruebas ayuda a las organizaciones a detectar diferencias entre las versiones del software, distinguiendo entre defectos y mejoras, y, como consecuencia, reducir el riesgo de enfrentar resultados inesperados y fallos funcionales al incorporar nuevo código. Su relevancia se basa en evaluar cambios, asegurar que no se introduzcan errores que impacten negativamente en el rendimiento del software, optimizar el tiempo y los costos en la generación de futuras versiones, y, como ya se mencionó, elevar la calidad del software [3]. En este sentido, en [4] se argumenta que la realización efectiva de las pruebas de regresión facilita la identificación de errores ya corregidos que pueden reaparecer debido a modificaciones en el código. Sin embargo, las pequeñas organizaciones desarrolladoras de software enfrentan dificultades importantes para realizar eficazmente estas pruebas, principalmente por restricciones de recursos y tiempo, y la ausencia de procesos bien establecidos.

La investigación presentada en [5], por ejemplo, mostró que incluso este tipo de organizaciones debe vigilar los efectos de realizar pequeñas modificaciones al código, porque pueden generarse errores que conduzcan a la alteración de funcionalidades clave del software. Sin embargo, realizar pruebas exhaustivas de regresión resulta costoso y poco práctico para estas organizaciones, lo que deriva en la falta de estrategias efectivas y en una dependencia excesiva del esfuerzo individual de los *testers*. Esto provoca inevitablemente altas tasas de fracaso, exceso de retrabajo y la generación de costos adicionales que perjudican la calidad del software y afectan negativamente el retorno de la inversión. A pesar de que en la actualidad existen diversos enfoques y métodos avanzados que emplean técnicas de la Inteligencia Artificial para clasificar, seleccionar, y/o priorizar aquellos casos de prueba que generen los mejores resultados en las regresiones, su uso aún es complicado en las pequeñas organizaciones debido a la falta de experticia, metodologías y herramientas apropiadas. Aunado a esto, los avances importantes que se han logrado en la automatización de las pruebas de regresión, principalmente en áreas operativas y organizativas, no se han enfocado en los aspectos estratégicos, como las políticas de adopción, gestión y medición, especialmente en el contexto de las pequeñas organizaciones desarrolladoras de software. En [6] se señala que la efectividad de una estrategia de pruebas de regresión depende de factores externos (como el tipo de producto y el cumplimiento normativo) e internos (como la filosofía de pruebas y la madurez de los equipos). Estos elementos condicionan decisiones clave, como la inversión en infraestructura y la gestión del cambio y los riesgos, aspectos que se realizan en las pequeñas organizaciones con mucha dificultad. Por otro lado, en [7] también se reconoce la complejidad que enfrentan las organizaciones para establecer estrategias efectivas y se sugieren enfoques que prioricen ciertos conjuntos de pruebas para agilizar su ejecución y aumentar la motivación de los *testers*. Aunado a lo anterior, en [8] se considera que la carencia de planificación y gestión en las pruebas de regresión también conduce a que las pequeñas organizaciones hagan un uso ineficiente de los recursos, ya que estas pruebas pueden representar hasta el 30% del esfuerzo total que se destina al aseguramiento de la calidad en la industria de software.

Por lo tanto, para superar estos obstáculos, es fundamental contar con herramientas tecnológicas y estrategias que faciliten a las pequeñas organizaciones la gestión y ejecución de las pruebas de regresión, además de que fomenten entre el personal un mejor entendimiento sobre su relevancia con el fin de asegurar la calidad del software

y minimizar los riesgos en los proyectos [9]. Esta idea coincide con los argumentos presentados en [10], puesto que se considera que el uso de enfoques mixtos podría facilitar la identificación de factores influyentes en cuanto a la adopción y aplicación de las pruebas de regresión dentro de una organización desarrolladora de software de pequeño tamaño. Sin embargo, es importante mencionar también, que estos mismos investigadores enfatizaron el hecho de que uno de los factores claves importantes para que esta idea funcione, consistía en considerar la madurez que coexiste dentro de la organización, lo que conlleva a definir prácticas eficaces en cuanto a las pruebas de regresión así como la coherencia que existe entre estas prácticas y la gestión tanto de la información (i.e., conocimiento) como del cambio en los procesos ya definidos en la organización [11].

Considerando todo lo anteriormente mencionado, el presente estudio presenta una estrategia de gestión que pretende mejorar la efectividad de las pruebas de regresión en el software que se desarrolla en las organizaciones de pequeño tamaño. En este contexto, antes de someter dicha estrategia a una evaluación empírica en una organización real, se realizó una evaluación cualitativa en la que participaron diferentes especialistas de la industria con el objetivo de evaluar su adherencia e identificar mejoras. El resto del artículo está organizado de la siguiente manera. La Sección II hace un análisis de las principales propuestas encontradas en la literatura para definir una estrategia que facilite la gestión y realización de las pruebas de regresión en el contexto de organizaciones desarrolladoras de software. La Sección III presenta la estrategia propuesta en el estudio, haciendo énfasis en los elementos que la conforman. En la Sección IV se presenta el diseño de la evaluación cualitativa y se analizan los resultados obtenidos. Finalmente, la Sección V presenta una discusión sobre los hallazgos encontrados y resume las principales conclusiones del estudio.

2. Estado del arte

Asegurar la calidad y funcionalidad del software representa un reto continuo, siendo las pruebas de regresión una práctica necesaria para lograrlo, ya que permiten verificar que las modificaciones en el código no generen errores ni perjudiquen las funcionalidades ya existentes. De acuerdo con [12], estas pruebas son particularmente importantes en entornos ágiles, caracterizados por iteraciones rápidas y frecuentes, ya que contribuyen a preservar la integridad del software al identificar defectos de manera temprana. No obstante, las pruebas de regresión pueden resultar costosas y demandantes para cualquier tipo de organización, especialmente en sistemas complejos con una gran cantidad de casos de prueba, lo que complica su planificación, ejecución y control.

Para enfrentar estos desafíos, se han llevado a cabo diversas investigaciones orientadas a optimizar y aumentar la eficiencia y efectividad de las pruebas de regresión en las organizaciones de desarrollo de software. Si bien, en la literatura especializada existe poca o nula evidencia de estrategias enfocadas al contexto de las pequeñas organizaciones, este análisis busca ofrecer una visión actualizada de las propuestas que pretenden formalizar y mejorar este proceso, independientemente del tamaño de la organización, destacando las mejores prácticas y áreas de mejora que impulsen el avance de la investigación y el aseguramiento de la calidad del software.

En este contexto, la investigación presentada en [13] mejoró la detección de errores en las pruebas de regresión mediante la creación de un método híbrido que combinara los métodos que se basan en el historial de pruebas y el contenido de los casos de prueba para recomendar casos de prueba adicionales. A pesar de que las recomendaciones de casos de prueba con mayor similitud fueron efectivas para detectar regresiones, se determinó que el éxito del método podría depender de las características del producto en general, del enfoque de las pruebas o incluso del conocimiento que poseen los *testers* para llevar a cabo las pruebas de regresión.

De manera similar, en [14] se presentó una estrategia que se basó en el análisis y creación de diferentes algoritmos para facilitar la elección del tipo de prueba más adecuado a un determinado contexto y en el diseño de un sistema de soporte a la toma de decisiones. Considerando razonamientos fundamentados en la lógica difusa y en metodologías de toma de decisiones multicriterio, se desarrolló una herramienta computacional que sirvió de apoyo a la toma de decisiones cuando se pretendía elegir los métodos de pruebas de regresión más adecuados para un proyecto de software específico. Sin embargo, el estudio no incluye una evaluación empírica en un contexto real, limitando su aplicabilidad práctica.

Por otro lado, en [15] se propuso el modelo llamado Réplica Optimizada del Recorrido de Hormiga León en Grafos Hash (OHGW-ALR, por sus siglas en inglés) para planificar los casos de prueba de acuerdo con su prioridad y, como consecuencia, predecir los fallos en cada regresión. Este enfoque logró organizar los casos de

prueba minimizando el uso de los recursos y el tiempo de procesamiento. En este sentido, de acuerdo con los datos presentados, el modelo OHGW-ALR obtuvo una buena tasa de predicción de fallos.

Finalmente, la investigación resumida en [16] considera que el uso de listas de verificación puede ayudar a los *testers* a supervisar las tareas clave en las pruebas de regresión, aportando organización al proceso. Con esta idea en mente, se realizó un estudio que hizo uso de entrevistas, talleres y cuestionarios para desarrollar listas de verificación que abordaran dos propósitos principales: (1) evaluar si los responsables y equipos de *testing* están listos para comenzar las pruebas de regresión, y (2) controlar las acciones críticas durante la planificación y ejecución de estas pruebas. El estudio incluyó a 25 especialistas de 12 empresas, quienes participaron en la creación y evaluación de dichas listas. Por último, todos los participantes consideraron que las listas propuestas son útiles y adaptables a sus entornos, y el 80% argumentó que cubren aspectos esenciales para las pruebas de regresión.

Como se puede apreciar, la mayor parte de los modelos, técnicas y estrategias descritas anteriormente han sido diseñadas para planificar, realizar y gestionar las pruebas de regresión en las organizaciones desarrolladoras de software, puesto que se asume un alto nivel de conocimiento, una amplia cantidad de recursos económicos y humanos, y una disponibilidad importante para ejecutar nuevos procesos. Sin embargo, estas estrategias a menudo no son factibles en el contexto de las pequeñas organizaciones, puesto que éstas enfrentan a menudo múltiples limitaciones que dificultan su implementación (e.g., recursos financieros, falta de capacitación y especialización del personal, infraestructura tecnológica, carencia de gestión y mantenimiento continuo). Por lo que, a continuación, se presenta una estrategia alternativa que pretende definir y formalizar un proceso que resulte efectivo para este tipo de organizaciones. Aunado a esto, antes de su implementación práctica, se siguió un enfoque cualitativo con el fin de comprender, a través de la observación y entrevista, las experiencias y perspectivas de los especialistas sobre dicha estrategia.

3. Estrategia propuesta

Aunque los métodos o técnicas que se encuentran en la literatura especializada para mejorar las pruebas de regresión han demostrado ser efectivos, su implementación y operación presentan barreras significativas para las pequeñas organizaciones. En este contexto, se identificó la necesidad de crear una estrategia potencialmente compatible con las necesidades y capacidades operativas de este tipo de organizaciones tomando en cuenta lo siguiente:

3.1. Bases conceptuales

1. Las recomendaciones presentadas en [16], donde se identifica un conjunto de actividades esenciales para antes y después de la realización de las pruebas de regresión, al adecuarlas al contexto de las pequeñas organizaciones desarrolladoras de software.
2. Los principios definidos en [17] y [18] para diseñar una estrategia basada en la gestión de conocimiento. Bajo esta premisa, no solamente se dotaría a la pequeña organización de una estrategia de gestión, sino que se inculcaría también en el personal la gestión de todo el conocimiento generado con la planeación, realización y gestión de las pruebas de regresión con el fin de mejorar la calidad de los productos generados.
3. El desarrollo de una herramienta computacional que considere las recomendaciones vertidas en [19] y [20], de tal manera que su integración en las pequeñas organizaciones no sea invasiva y les permita madurar el proceso relacionado con la realización de las pruebas de regresión.

3.2. Descripción de la PAL

Dadas las características de las pequeñas organizaciones desarrolladoras de software, se propone la creación de una Biblioteca de Activos (PAL, por sus siglas en inglés) para definir el proceso de pruebas. La PAL representa el medio necesario para definir, realizar, gestionar y medir la realización de las pruebas de regresión en una pequeña organización, abordando la complejidad intrínseca de este proceso y dando soporte, al mismo tiempo, al personal involucrado en aspectos clave para la realización exitosa de este tipo de pruebas. De acuerdo con [21], al seguir esta estrategia se les proporciona a las pequeñas organizaciones el conocimiento esencial para definir, establecer y difundir su proceso de pruebas de regresión. Por consiguiente, la PAL es un repositorio común de

datos (i.e., activos) que permite la estandarización de procesos dado que todos los proyectos pueden utilizar el mismo proceso estándar o diferentes adaptaciones aprobadas de este. Es decir, un proceso estándar para las pruebas de regresión que se encuentre contenido en la PAL se define en términos de actividades, productos, medidas (de proceso, actividades y productos) y activos (también de proceso, actividades y productos). Todos estos activos son artefactos (e.g., plantillas, guías de llenado, políticas de uso) o mecanismos (e.g., matriz de trazabilidad, petición de cambios) que brindan soporte para realizar el proceso de pruebas de regresión, llevar a cabo las actividades, obtener los productos, y recoger la información relacionada con las medidas.

Además, para cada proyecto que se realice dentro de la organización, se podrá definir, a partir de la PAL, un proceso específico que considere los criterios y pautas de adaptación. De esta manera, el proceso que se defina en la organización para realizar y gestionar las pruebas de regresión podrá ser utilizado en diferentes proyectos promoviendo así la actualización del conocimiento contenido en la PAL en dos niveles: (i) a nivel de proceso, producto y tarea con medidas reales, y (ii) a nivel de mejora con lecciones aprendidas (i.e., el conocimiento adquirido a través de la reflexión sobre experiencias, tanto positivas como negativas, durante el desarrollo de un proyecto de software) y las mejoras identificadas.

La Figura 1 describe la manera en que la estrategia propuesta hace uso de la PAL para generar una cultura de trabajo diferente en el contexto de una pequeña organización. En esta figura, los procesos que son definidos a partir del proceso estándar para las pruebas de regresión (ya contenido en la PAL) se adaptarán a cada proyecto en función de los activos específicos que sean seleccionados.

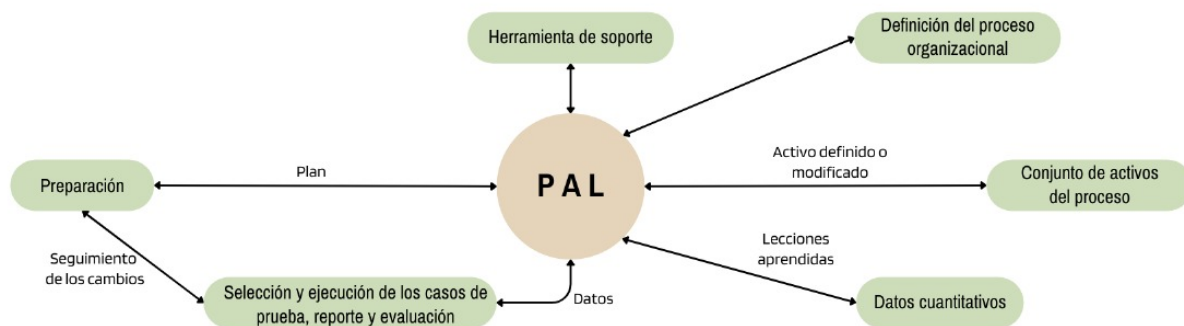


Figura 1. PAL para mejorar la efectividad de las pruebas de regresión en las pequeñas organizaciones.

3.3. Estructura de la PAL

En este sentido, los procesos que se adapten a cada proyecto se podrán diferenciar dependiendo de las relaciones de los elementos mostrados en la Figura 1 y los activos escogidos, como se describe a continuación:

- Cualquier cambio que se realice al código fuente requiere de un análisis cuidadoso para realizar las pruebas de regresión. Dicho análisis se realiza en la fase de Preparación, donde los cambios son documentados y gestionados mediante el control de versiones. De esta manera, la PAL facilita la planificación y el seguimiento y control de los cambios que deberán ser probados en la correspondiente regresión.
- Posteriormente, en la fase de Selección y ejecución de los casos de prueba, reporte y evaluación se definen y utilizan los casos de prueba que deberán abordar los cambios realizados sobre el código. Posteriormente, y con los datos cuantitativos recogidos en el proyecto, estos casos de prueba pueden ser priorizados y/o categorizados dependiendo su desempeño en el proyecto. Esta información será útil para que la pequeña organización mejore su madurez con base en la experiencia, puesto que aquellos casos de prueba que conduzcan a una buena detección de defectos y/o fallos serán identificados y, como consecuencia, representarán a un conjunto ideal para la reutilización.
- Dado que la PAL introduce la definición y gestión del conocimiento generado durante la realización de las pruebas de regresión, se requiere que la pequeña organización siga utilizando las herramientas de soporte (manuales o automatizadas) con las que habitualmente realiza este tipo de pruebas.

- Uno de los propósitos fundamentales de la PAL es que el conjunto de activos que conforman al proceso evolucione a través de su uso repetido. Es importante recalcar que la estructura con la que fue diseñada la PAL incorpora un conjunto base de activos que deberá evolucionar con el paso del tiempo.
- Finalmente, la evolución de los activos conducirá a la creación de nuevos procesos de pruebas de regresión a partir del proceso organizacional (i.e., proceso estándar de la organización), el cual servirá como guía. Así, cada proyecto que involucra a los activos e información contenida en la PAL generará un conjunto de lecciones aprendidas que permitirá la mejora continua de la organización.

Considerando esta estructura de esta PAL, se diseñó una estrategia que la integrara para definir y mejorar la forma en que las pequeñas organizaciones realizan las pruebas de regresión (ver Figura 2).

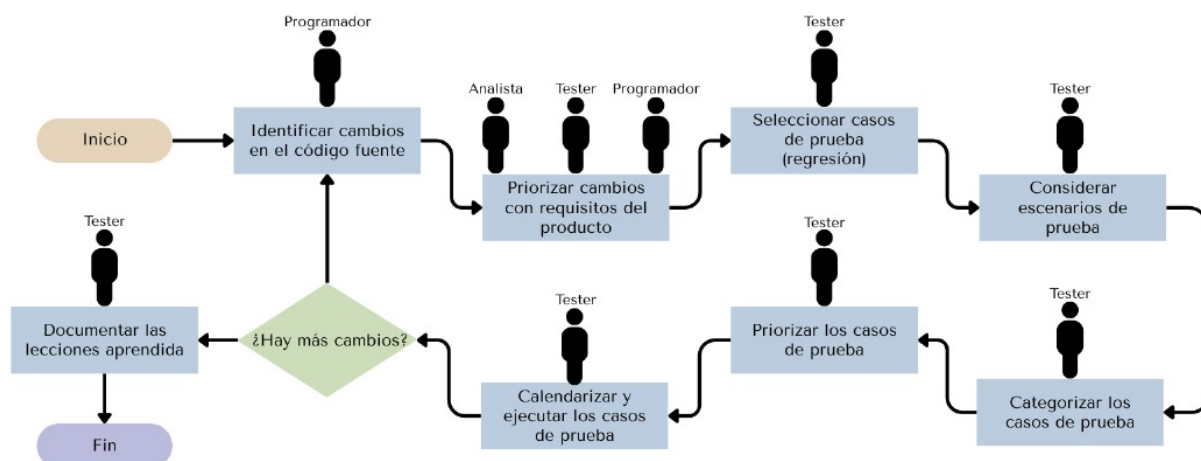


Figura 2. Estrategia alternativa para definir, ejecutar y gestionar las pruebas de regresión en las pequeñas organizaciones de software.

3.4. Proceso para realizar las pruebas de regresión

El proceso establecido por la estrategia inicia con la identificación de los cambios que se hayan realizado al código fuente, lo cual se denomina formalmente como “regresión”. De acuerdo con [5], es común que las decisiones que se toman en las pequeñas organizaciones se basen en la experiencia previa, que la mayoría de las veces suele estar poco o nada relacionada con el contexto de un proyecto específico. Por lo tanto, se pretende inculcar inicialmente en estas organizaciones la cultura de documentación eficiente con el fin de crear un acervo digital de información histórica que no solamente les permita generar una base reutilizable de conocimiento, sino que también les facilite la mejora continua de sus prácticas cotidianas y, como consecuencia, el madurar sus procesos de software.

Así, la primera actividad implica que la organización defina y mantenga actualizada una matriz de trazabilidad (i.e., la herramienta que establece relaciones entre diferentes elementos de un proyecto de software, como requisitos, casos de prueba, diseño y entregables) con el fin de que las modificaciones que se realicen sobre el código estén documentadas. Posteriormente, antes de que el programador modifique el código deberá realizar peticiones de cambio con el fin de que éstas sean evaluadas por el jefe de proyectos y se tome una decisión informada.

Es importante mencionar que todas las actividades definidas por el proceso mostrado en la Figura 2 son soportadas por activos que guían puntualmente a los involucrados para asegurar su correcta realización (ver Figura 3). Todo activo se acompaña de una guía de llenado, siguiendo el buen ejemplo que dieron modelos como el *Personal Software Process* y *Team Software Process* del Instituto de Ingeniería de Software de la Universidad de Carnegie Mellon.

Posteriormente, el analista, tester y programador deberán analizar los cambios que se hicieron al código con la intención de identificar qué requisitos del producto están involucrados en la modificación a fin de priorizar qué se probará primero. Por lo tanto, esta actividad involucra tanto al analista como al tester y al jefe de proyectos, puesto que es probable que la modificación realizada origine una modificación tanto en los requisitos del producto de software como en los casos de prueba que fueron creados durante el desarrollo de este, lo que conducirá a que se incremente la cantidad de casos de prueba en el repositorio del proyecto.

En este sentido, la estrategia plantea que la pequeña organización considere la probabilidad de ocurrencia de un fallo y el impacto que puede ocasionar la modificación al producto de software. Así, se le proporciona al *tester* la guía adecuada para que realice un análisis cualitativo/cuantitativo de las modificaciones, considerando su severidad, para determinar qué casos de prueba se ejecutarán nuevamente o bien si es necesario crear nuevos a fin de lograr una mayor cobertura.

Una vez que la priorización de los cambios sobre los requisitos se ha llevado a cabo, el *tester* deberá seleccionar qué casos de prueba ejecutará nuevamente o bien si es necesario crear nuevos casos de prueba para tener mayor cobertura. Con el objetivo de no complicar este proceso a las pequeñas organizaciones, se plantea que la selección de los casos de prueba se base en dos aspectos fundamentales: (1) la severidad de los fallos relacionados con los requisitos funcionales involucrados en las modificaciones al código y (2) la repetitividad de un fallo en los casos de prueba. Con relación al segundo aspecto, se ha creado la definición de una categoría para los fallos que pueden originarse durante la ejecución de los casos de prueba con el fin de que el *tester* de la pequeña organización pueda documentarlos y, como consecuencia, sea capaz de determinar la cantidad de veces que un fallo está repitiéndose en los casos de prueba. Considerando lo anterior, el *tester* podrá identificar y seleccionar aquellos casos de prueba que deberá considerar inicialmente para ser ejecutados a causa de las modificaciones realizadas al código. Es importante hacer notar que, dado que se trata de un proceso cíclico, el conjunto de casos de prueba irá aumentado con cada regresión y que, incluso, es posible que se deban agregar nuevos requisitos, lo que implicaría generar también nuevos casos de prueba.

Solicitud de cambios: Activo SOLCA				
			Fecha:	
Nombre del producto:			Referencia:	
Nombre del proyecto:				
Nombre del jefe de proyecto:				
Nombre del programador (quien solicita el cambio):				
Descripción del cambio solicitado:				
Justificación del cambio solicitado:				
Análisis de impacto				
Cronograma estimado para la mejora del producto:				
Alcance (funcionalidad del producto):				
Presupuesto destinado a la mejora del producto:				
¿El cambio solicitado está justificado para llevarse a cabo?			Sí	No
Justificación:				
Aprobación de la solicitud				
Fecha de aprobación/rechazo:				
Nombre y firma del jefe de proyecto		Nombre y firma del programador solicitante		
Nombre		Nombre		

Figura 3. Ejemplo de activo para implementar la estrategia.

Con la intención de formalizar la prueba, el *tester* deberá documentar descripciones de los casos de prueba (escenarios) que le permitan validar que el software cumple con los requisitos tal y como se espera. Así pues, bajo este razonamiento, el *tester* deberá simular situaciones de la vida real y observar cómo interactúan los usuarios con el software mientras realizan acciones específicas para verificar que este hace lo que debería y no lo que no debería. En el contexto de la estrategia propuesta, se orienta al *tester* para que simplifique esta labor a través de la creación de escenarios positivos y negativos. En este sentido, se le indica cómo utilizar el Enfoque Guiado en el Comportamiento (BDD, por sus siglas en inglés) [22] para describir los escenarios. Los escenarios creados servirán

ahora para que el *tester* genere casos de prueba que cubran las funcionalidades modificadas y/o agregadas por el programador. La estrategia le indica al *tester* categorías que deberá utilizar para clasificar cada caso de prueba. Con el término de la categorización de los casos de prueba, el *tester* seleccionará y priorizará aquellos casos que serán ejecutados con el fin de gestionar el avance sobre la profundidad de las pruebas. Por lo tanto, la estrategia propuesta plantea que la pequeña organización se base principalmente en la priorización basada en la probabilidad de ocurrencia y el impacto, o bien lo que se definió como severidad para priorizar los requisitos que debían modificarse con la regresión. De esta manera, y considerando lo mencionado anteriormente, el *tester* puede determinar qué tan críticos y esenciales son los casos de prueba tanto para el negocio como para el producto de software.

Siguiendo con el proceso, el *tester* planificará la ejecución de los casos de prueba seleccionados con el fin de gestionar el avance de sus actividades. Es importante considerar que esta planificación será independiente del plan de proyecto original. Es decir, una vez que los casos de prueba con las modificaciones más relevantes y recientes hayan sido seleccionados por el *tester*, debe calendarizar su ejecución. Para tales efectos, la estrategia propuesta guía al *tester* en la ejecución de tres tareas básicas: preparación, ejecución y cierre.

Cuando el *tester* haya agotado toda la lista de casos de prueba, deberá evaluar si existen cambios que se hayan agregado para abordarlos de nuevo. En caso contrario, el producto se libera y se procede a cerrar las pruebas de regresión para ese producto en específico.

Finalmente, con el fin de proceder al cierre del proceso mostrado en la Figura 2, el *tester* deberá documentar las lecciones aprendidas con el fin de contribuir a la generación de conocimiento que ayude a la organización a no cometer errores innecesarios en proyectos futuros. Para esto, la estrategia establece una clasificación de incidencias que se deberá considerar al documentar cada lección (i.e., activo, comunicación, deficiencia, factor de éxito, gestión, liderazgo, persona, problema, reutilización, refactorización, técnica). Por lo tanto, la estrategia establece que este proceso se repita hasta que las regresiones hayan sido terminadas y se cuente con un producto mejorado de software que pueda ser utilizado para cubrir nuevas necesidades de negocio.

Una vez que se finalizó el diseño de la estrategia, y antes de crear una herramienta computacional que facilite su implementación en las pequeñas organizaciones, se procedió a evaluar su adherencia a través de una evaluación subjetiva de las fases, actividades, activos, medidas, y guías contenidos en la PAL. A continuación, se resumen los resultados obtenidos.

4. Evaluación cualitativa de la adherencia de la estrategia

Se diseñó un estudio bajo un enfoque cualitativo con la finalidad de analizar y entender a detalle las percepciones de los profesionales de la industria de software sobre la estrategia presentada en la sección anterior. En este sentido, se decidió seguir un enfoque cualitativo por su capacidad para percibir las opiniones, experiencias y dinámicas humanas en un entorno organizacional [23], obteniendo así una percepción puntual sobre la manera en que la estrategia propuesta podría impactar en la mejora de la efectividad de las prácticas de desarrollo de software de las pequeñas organizaciones. Como se mencionó anteriormente, de esta manera se pretendió determinar el grado de adherencia de la estrategia con el fin de hacer las adecuaciones necesarias y crear, en una segunda fase del estudio, una herramienta computacional que implemente una versión refinada de la misma. En las siguientes secciones se presenta el diseño de esta evaluación.

4.1. Diseño del estudio diagnóstico

El estudio fue de tipo exploratorio y descriptivo, con el objetivo de examinar las opiniones de los profesionales sobre la estrategia propuesta. Este enfoque facilita la identificación de patrones y tendencias de uso en un contexto organizacional sin que el investigador se involucre directamente. En este sentido, se comenzó por compartir a través de un repositorio en la nube toda la información correspondiente a la estrategia para que fuera revisada por los participantes en la evaluación, incluyendo la descripción detallada del proceso mostrado en la Figura 2 a nivel de fases y actividades, definición de medidas, guías de adaptación, políticas de uso, activos, y guías de llenado. La información estuvo disponible por 30 días para promover una revisión escrupulosa. Posteriormente, se diseñó y aplicó un cuestionario en línea para recoger información sobre las percepciones de los participantes sobre la estructura, elementos y utilidad de la estrategia compartida. Dicho cuestionario fue revisado previamente por expertos en el área de pruebas de software con el objetivo de obtener retroalimentación que mejorará este

instrumento de evaluación. Por último, se llevaron a cabo entrevistas semiestructuradas con el objetivo de escudriñar en las percepciones y discrepancias de los participantes respecto a los elementos incorporados en la estrategia para mejorar el proceso de pruebas de regresión.

4.2. Selección de los participantes

La elección de los participantes se llevó a cabo a través de un muestreo intencional, enfocado a involucrar a profesionales con roles específicos dentro de sus organizaciones (i.e., jefes de proyectos, programadores y *testers*). Se invitó a 52 micro y/o pequeñas organizaciones desarrolladas de software a participar en el estudio, de las cuales 31 respondieron positivamente. La muestra incluyó a 135 profesionales de organizaciones de los estados de Baja California del Norte, Ciudad de México, Oaxaca, Puebla, Querétaro y Tlaxcala. Estos participantes, con una antigüedad promedio de cinco años, aportaron información valiosa para valorar la adherencia de la estrategia a sus respectivos entornos organizacionales.

4.3. Recolección de datos

Los datos fueron recolectados de dos maneras: (1) a través de la aplicación remota de un cuestionario que fue diseñado para recoger las percepciones de los profesionales sobre la aplicabilidad de la estrategia en sus organizaciones (ver Tabla 1) y (2) mediante entrevistas semiestructuradas que se realizaron tanto de forma presencial como virtual, dependiendo de la disponibilidad de los participantes y las condiciones logísticas del estudio.

Tabla 1. Cuestionario para recoger las percepciones de los participantes sobre la estrategia propuesta.

#	Ítem	Respuestas					M	DE
		TA	DA	N	D	TD		
1	La documentación que revisé, relacionada con la estrategia para definir un proceso para realizar y gestionar las pruebas de regresión, es fácil de entender.	62	50	15	8	0	1.2	0.9
2	El lenguaje utilizado para describir todos los elementos de la estrategia es el adecuado.	42	93	0	0	0	1.3	0.5
3	Considero que esta estrategia se puede adaptar fácilmente a las necesidades de una pequeña organización desarrolladora de software.	42	45	12	21	15	0.6	1.4
4	El proceso definido para realizar y gestionar las pruebas de regresión podría adecuarse al contexto de mi organización.	97	28	10	0	0	1.6	0.6
5	La adopción de este nuevo proceso alteraría considerablemente nuestra cultura de trabajo con relación a las pruebas de regresión.	127	8	0	0	0	1.9	0.2
6	El éxito en la adopción del nuevo proceso dependerá, de manera importante, de la capacitación que se reciba.	38	97	0	0	0	1.3	0.4
7	Las fases y actividades del proceso descrito para realizar y gestionar las pruebas de regresión son descritas con el nivel de detalle necesario.	79	38	18	0	0	1.5	0.7
8	Los diagramas que acompañan la descripción detallada de las actividades son claros y representativos del trabajo que se requiere realizar.	90	42	3	0	0	1.6	0.5
9	Los activos de la PAL están definidos de tal manera que se facilita considerablemente su uso en mi organización.	4	116	11	4	0	0.9	0.5
10	La cantidad de activos creados para facilitar la implementación del proceso es suficiente para realizar y gestionar las pruebas de regresión.	134	1	0	0	0	2.0	0.1
11	Las guías de llenado que acompañan a los activos son claras y fáciles de seguir.	128	5	2	0	0	1.9	0.3
12	Considero que en mi organización se cuenta con las habilidades (suaves y duras) necesarias para implementar este tipo de procesos.	84	35	16	0	0	1.5	0.7

#	Ítem	Respuestas					M	DE
		TA	DA	N	D	TD		
13	Entendí fácilmente las guías de adaptación para adecuar un proceso institucional a diferentes proyectos que requieran la realización de las pruebas de regresión.	0	5	46	63	21	-0.7	0.8
14	Estoy seguro que con la orientación adecuada, yo y mis compañeros de equipo podríamos seguir puntualmente la estrategia sin problema alguno.	123	12	0	0	0	1.9	0.3
15	Estoy convencido de que una herramienta computacional simplificaría la implementación de la estrategia en mi organización.	89	46	0	0	0	1.7	0.5

Cada sesión de trabajo tuvo una duración aproximada de 45 a 60 minutos y fue grabada en audio, contando con la previa autorización de los entrevistados. El cuestionario se estructuró para explorar, a través de 15 ítems, aspectos como el ajuste de la estrategia a la cultura de trabajo de la organización, la capacidad del personal para realizar las actividades establecidas por la estrategia, la facilidad de uso de los activos, la claridad de las guías de adaptación y llenado, y la formalidad en la definición del proceso para realizar y gestionar las pruebas de regresión. Dicho instrumento se basó en una escala *Likert* de 5 puntos con las siguientes alternativas de respuesta: “Totalmente de acuerdo” (TA=2), “De acuerdo” (DA=1), “Neutral” (N=0), “En desacuerdo” (ED=-1) y “Totalmente en desacuerdo” (TD=-2), que se convirtieron a un valor numérico para obtener una medida cuantitativa. Además, se calcularon la mediana (M) y la desviación estándar (DE) de las respuestas proporcionadas por todos los participantes en la evaluación. Finalmente, el análisis de las entrevistas se realizó con el método de codificación temática para identificar, analizar y comprender las ideas y perspectivas presentes en las respuestas de los participantes.

4.4. Consideraciones éticas

El cumplimiento de los principios éticos de la investigación fue asegurado mediante la protección de la confidencialidad y el anonimato de los participantes. Previo a su inclusión en el estudio, cada participante firmó un documento de consentimiento informado donde se especificaban los objetivos del estudio, el tratamiento que se daría a los datos y la libertad de retirarse del proceso cuando lo consideraran oportuno. Además, toda la información recopilada se almacenó en la nube con un acceso limitado.

4.5. Análisis de datos y hallazgos obtenidos

La información recopilada fue transcrita y examinada mediante el análisis descriptivo, lo que ayudó a detectar opiniones recurrentes, tendencias y discrepancias relevantes en las respuestas proporcionadas por los encuestados. Para garantizar la validez del estudio se aplicó la triangulación de datos, contrastando las transcripciones y los resultados con los mismos participantes. Analizando la información recogida con el cuestionario se observó que la mayoría de los participantes consideró que la documentación compartida sobre la estrategia es clara y, por ende, fácil de entender (M=1.2, DE=0.9). Sin embargo, el 17.0% no lo considera así. En la entrevista estos participantes argumentaron que (1) era difícil determinar un entendimiento adecuado al revisar únicamente descripciones, diagramas y plantillas y (2) que su opinión probablemente estuviera sesgada por la falta de una herramienta real. Vale la pena mencionar que se determinó que el 85.0% de estos participantes tenían la formación de ingenieros electrónicos y mecatrónicos, por lo que una falta de formación en el proceso de pruebas pudo haber sesgado también su opinión.

A pesar de la opinión anterior, el 100% de los participantes opinó que el lenguaje utilizado para desarrollar la estrategia era el adecuado (M=1.3, DE=0.5). La ausencia de ambigüedad en las respuestas demuestra que el lenguaje empleado en la estrategia no genera confusión y, de acuerdo con los encuestados, puede favorecer a una implementación fluida.

Aunado a esto, el 36.0% de los encuestados consideró que la adaptación de la estrategia al contexto de las pequeñas organizaciones no sería fácil (M=0.6, DE=1.4). Al indagar sobre esta percepción, los profesionales tuvieron diferentes puntos de vista: algunos consideraron que, si la estrategia se basaba únicamente en descripciones textuales y plantillas, representaría más trabajo burocrático; otros respondieron no estar seguros de

que en sus organizaciones tendrían el tiempo para generar plantillas; y un grupo más pequeño se mostró inseguro de seguir correctamente lo indicado en la estrategia debido a su formación académica. Esta información será de mucha utilidad más adelante para seguir estrategias adecuadas que reduzcan el rechazo cultural de la mejora.

Al ser más específicos en la implementación de la estrategia en sus organizaciones, los participantes dieron opiniones totalmente diferentes, puesto que el 92.0% consideró que el proceso descrito por la estrategia podría adecuarse a su entorno de trabajo ($M=1.6$, $DE=0.6$). Al indagar en las entrevistas, el resto de los participantes argumentó, principalmente, que su opinión negativa se debía al desconocimiento de si en su organización habría disposición por cambiar la forma de trabajo.

Por otro lado, el 100% de los encuestados consideró que la implementación de un proceso diferente para realizar y gestionar las pruebas de regresión alteraría considerablemente la cultura de trabajo de su organización ($M=1.9$, $DE=0.2$). No obstante, al indagar en la entrevista, el 93.0% consideró que sería un cambio positivo.

Además, la mayoría de los participantes coincidió en que el éxito en la adopción del nuevo proceso de pruebas de regresión dependería significativamente de la capacitación que se reciba ($M=1.3$, $DE=0.4$). Esto refleja un reconocimiento colectivo de que, si bien, la estrategia descrita es clara y estructurada, su implementación requerirá el desarrollo de competencias prácticas y una comprensión a detalle de los flujos de trabajo para realizar la priorización de los cambios, la selección de los casos de prueba o la documentación de las lecciones aprendidas.

A pesar de que a la mayoría de los participantes le pareció clara y fácil de entender la documentación de la estrategia (diagramas, descripciones y plantillas), el 17.0% no consideró que las fases y actividades estuvieran descritas con el nivel de detalle necesario ($M=1.5$, $DE=0.7$), principalmente por dos razones: si bien el proceso y las descripciones son comprensibles, estos recursos por sí solos no les permiten adquirir un entendimiento adecuado de cómo se ejecutarían las pruebas en la práctica, y la falta de una validación práctica generó opiniones negativas.

Los participantes, en su mayoría, también consideraron que los diagramas asociados a la descripción de las actividades eran claros y representaban adecuadamente al flujo de trabajo establecido por la estrategia ($M=1.6$, $DE=0.5$). Sin embargo, tres de los profesionales que participaron el estudio no respondieron afirmativamente argumentando que las actividades descritas no se alinean con sus funciones y áreas de experticia. Como dato adicional, se trató de profesionales con formación de Electrónica y Mecatrónica que realizan las pruebas de regresión.

El 89.0% de los participantes consideró que los activos de la PAL fueron definidos de manera clara facilitando su uso en la organización ($M=0.9$, $DE=0.5$). Sin embargo, el resto de los participantes no compartió este punto de vista por dos razones principales: (1) la dificultad de asegurar un entendimiento de la estrategia basándose únicamente en material descriptivo, sin ejemplos prácticos, y (2) la falta de validación mediante una aplicación real que demuestre la utilidad de los activos. En la entrevista se corroboró que los perfiles de estos profesionales están más orientados a la realización de pruebas sobre sistemas de hardware y embebidos. No obstante, esta información resalta la importancia de complementar los activos con casos prácticos o capacitaciones específicas para asegurar una adopción uniforme en equipos multidisciplinarios.

De manera similar, el 100% de los encuestados coincidió en que la cantidad de activos era suficiente para lograr la implementación del proceso ($M=2.0$, $DE=0.1$), mientras que el 98% consideró que las guías de llenado que acompañan a estos activos eran claras y fáciles de seguir ($M=1.9$, $DE=0.3$), lo que demuestra que estos materiales complementarios logran su objetivo de orientar de manera efectiva a los profesionales en la implementación del proceso de pruebas de regresión. Continuando con el análisis, el 88.0% de los participantes consideró que su organización posee las habilidades técnicas (duras) y personales (suaves) necesarias para implementar con éxito el proceso de pruebas de regresión descrito en la estrategia ($M=1.5$, $DE=0.7$). En contraste, el 12.0% que manifestó inseguridad en su respuesta argumentó durante la entrevista que requerirían mayor capacitación en aspectos específicos del flujo de trabajo. En conjunto, los resultados indican que la organización cuenta con una base sólida de competencias, aunque con margen para fortalecer ciertos aspectos que impulsen una implementación más robusta e inclusiva.

Los resultados revelaron un desafío importante para lograr la comprensión de las guías de adaptación para implementar el proceso de pruebas de regresión en diferentes proyectos. Mientras que solo el 4.0% de los participantes afirmó entenderlas fácilmente, el 62.0% expresó dificultades claras y el 34% permaneció indeciso ($M=0.7$, $DE=0.8$). Durante las entrevistas se confirmó que, aunque la estrategia puede ser clara, las guías específicas para adaptarla a contextos diversos no están logrando transmitir su propósito de manera efectiva. Los altos porcentajes de respuestas negativas y neutras revelan que las instrucciones podrían carecer de ejemplos

específicos, facilidad para distintos tipos de proyectos o claridad. Por lo que será necesario que, una vez que se cuente con una herramienta computacional, se capacite a este tipo de profesionistas para que, a través de la práctica, comprendan la capacidad de adaptación de la estrategia.

Un número notable de participantes (91.0%) argumentó sentirse totalmente convencidos de que, con la orientación adecuada, podrían seguir de manera puntual la estrategia presentada en este estudio ($M=1.9$, $DE=0.3$). Este consenso muestra que los profesionales experimentaron confianza tanto del proceso descrito en la estrategia como de la capacidad del personal de su organización para implementarlo con el apoyo necesario.

Finalmente, la totalidad de los participantes coincidió en que una herramienta computacional facilitaría significativamente la implementación de la estrategia de pruebas de regresión en su organización ($M=1.7$, $DE=0.5$). En este sentido, los participantes consideraron que es necesario automatizar la estrategia propuesta con el fin de agilizar la adopción de tareas como la identificación de cambios en el código, la priorización de pruebas o la documentación de los resultados. Aunado a lo anterior, la mayoría de los profesionales argumentaron que una limitante en la mejora de los procesos es la definición de métodos manuales puesto que dificulta la adopción de nuevas prácticas. Por lo tanto, estas opiniones indican un sólido respaldo para considerar el desarrollo o la adquisición de herramientas computacionales como paso siguiente en la optimización del proceso de pruebas de regresión.

La información recogida con estos hallazgos cualitativos permitió la mejora de la estrategia a nivel de fases, actividades, activos y medidas y actualmente se está trabajando en el desarrollo de la herramienta computacional que será evaluada empíricamente en las pequeñas organizaciones que deseen seguir siendo parte del estudio.

4.6. Limitaciones del estudio

Por último, vale la pena mencionar que este estudio presenta algunas limitaciones propias del enfoque cualitativo, como el número reducido de participantes y la subjetividad implícita de sus respuestas al cuestionario. En este sentido, si bien los resultados ofrecen las percepciones exactas de los profesionales de diversas pequeñas organizaciones desarrolladoras de software sobre la estrategia resumida en la Sección 3, las conclusiones no pueden extrapolarse a otras organizaciones o entornos pequeños con culturas diferentes de trabajo. No obstante, se intentó compensar el efecto de estas restricciones mediante el uso de la triangulación metodológica y la revisión minuciosa de las transcripciones y los análisis realizados.

5. Discusión y conclusiones

En el presente estudio se presentaron los resultados de realizar una evaluación cualitativa con la finalidad de analizar y entender a detalle las percepciones de 135 profesionales de la industria de software sobre una estrategia diseñada para definir un proceso que facilite la realización y gestión de las pruebas de regresión en las pequeñas organizaciones desarrolladoras de software. En este sentido, se decidió seguir un enfoque cualitativo por su capacidad para percibir las opiniones, experiencias y dinámicas humanas en un entorno organizacional, obteniendo así una percepción puntual sobre la manera en que la estrategia propuesta podría impactar en la mejora de la efectividad de las prácticas de desarrollo de software de este tipo de organizaciones. La información recogida condujo a la obtención de las siguientes conclusiones:

- El grado de adherencia de la estrategia es apropiado para realizar una implementación práctica. Si bien existen algunos temores en los profesionales, debido principalmente por su formación e incertidumbre si una pequeña organización estaría dispuesta a cambiar, también existe disposición e interés por mejorar la forma de trabajo.
- Las percepciones de los profesionales han facilitado la identificación de las adecuaciones necesarias para diseñar una herramienta computacional que implemente una versión refinada de la estrategia.
- Existe un interés latente de las organizaciones participantes en el estudio para involucrarse en la evaluación empírica de la herramienta.

Aunado a lo anterior, si bien los resultados obtenidos de la evaluación cualitativa presentada en este estudio coinciden con los argumentos presentados en [5], [6], [7] y [10] sobre la importancia de definir, realizar y gestionar apropiadamente las pruebas de regresión en las pequeñas organizaciones de forma efectiva y asequible, han permitido además identificar y trazar una estrategia que, en base a una PAL, permitirá mejorar la realización

de las pruebas de regresión en estas organizaciones. De esta manera, se ha propuesto una estrategia que se pretende ajustar a las necesidades de las pequeñas organizaciones, la cuales, con apoyo tecnológico, podrán definir, realizar y gestionar efectivamente este tipo de pruebas. En breve se evaluará empíricamente la efectividad del enfoque completo con el fin de contribuir a la mejora continua en la realización y gestión de las pruebas de regresión en las pequeñas organizaciones desarrolladoras de software y, como consecuencia, el desarrollo de productos de software de calidad.

6. Agradecimientos

Los autores agradecen el apoyo que la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) ha otorgado para la realización de este estudio a través de su convocatoria de becas nacionales para estudios de posgrado.

7. Referencias

- [1] Qasim, M., Bibi, A., Hussain, S. J., Jhanjhi, N. Z., Humayun, M., Sama, N. U. (2021). Test case prioritization techniques in software regression testing: An overview. *International Journal of Advanced and Applied Sciences*, 8 (5), 107-121. <https://doi.org/10.21833/ijaas.2021.05.012>
- [2] Thota, M. K., Shajin, F. H., Rajesh, P. (2020). Survey on software defect prediction techniques. *International Journal of Applied Science and Engineering*, 17 (4), 331-344. [https://doi.org/10.6703/IJASE.202012_17\(4\).331](https://doi.org/10.6703/IJASE.202012_17(4).331)
- [3] Kandukuri, P. (2020). *Regression testing: A model driven approach*. Lambert Academic Publishing.
- [4] Bertolino, A., Guerriero, A., Miranda, B., Pietrantuono, R., Russo, S. (2020). *Learning-to-rank vs ranking-to-learn: Strategies for regression testing in continuous integration*. ACM/IEEE 42nd International Conference on Software Engineering. Seoul, South Korea. <https://doi.org/10.1145/3377811.3380369>
- [5] Tuape, M., Hasheela-Mufeti, V. T., Kayanda, A., Porras, J., Kasurinen, J. (2021). Software engineering in small software companies: Consolidating and integrating empirical literature into a process tool adoption framework. *IEEE Access*, 9, 130366-130388. <https://doi.org/10.1109/ACCESS.2021.3113328>
- [6] Greca, R., Miranda, B., Bertolino, A. (2023). State of practical applicability of regression testing research: A live systematic literature review. *ACM Computing Surveys*, 55 (13s), 1-36. <https://doi.org/10.1145/3579851>
- [7] Govil, N., Sharma, A. (2021). *A game plan to build optimized regression testing in agile methodologies using test prioritization*. IEEE 5th International Conference on Information Systems and Computer Networks (ISCON). Mathura, India. <https://doi.org/10.1109/ISCON52037.2021.9702479>
- [8] Figueredo, L. (2021). Proceso de pruebas de software para un modelo de calidad en Cuba. *Revista de I+ D Tecnológico*, 17 (1), 23-35. <https://doi.org/10.33412/idt.v17.1.2914>
- [9] Kulkarni, V., Mohan, R. M., Reddy, H. V. (2021). Regression test optimization and automation in agile framework: A review. *Turkish Journal of Computer and Mathematics Education*, 12 (12), 2852-2856. <https://turcomat.org/index.php/turkbilmate/article/view/7954>
- [10] Kandil, P., Moussa, S., Badr, N. (2017). Cluster-based test cases prioritization and selection technique for agile regression testing. *Journal of Software: Evolution and Process*, 29 (6). <https://doi.org/10.1002/smr.1794>
- [11] Abdel Razek, R. A., Mahmoud Nagdy, N., Tharwat, G., El-Aziz Kolkila, A. A. (2024). Software testing using cutting-edge technologies with supporting knowledge management: A survey. *Journal of Al-Azhar University Engineering Sector*, 19 (72), 15-30. <https://doi.org/10.21608/aej.2024.248722.1471>
- [12] Das, S., Gary, K. (2025). Regression testing in agile—A systematic mapping study. *Software*, 4 (2), 1-19. <https://doi.org/10.3390/software4020009>
- [13] Aman, H., Nakano, T., Ogasawara, H., Kawahara, M. (2018). *A topic model and test history-based test case recommendation method for regression testing*. IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW). Sweden. <https://doi.org/10.1109/ICSTW.2018.00079>
- [14] Chernov, V., Dorokhova, L., Dorokhov, O., Egorovskaya, G. (2019). Decision support system choosing software testing strategy. *Bulletin of the Transylvania University of Brasov. Series III: Mathematics and Computer Science*, 12 (61), 457-468. <https://doi.org/10.31926/but.mif.2019.12.61.2.23>

- [15] Sivaji, U., Rao, P. S. (2021). Improving regression testing query replying procedure using secure optimized graph walk scheme. *Journal of Theoretical and Applied Information Technology*, 99 (9), 2093-2103. <https://www.jatit.org/volumes/Vol99No9/14Vol99No9.pdf>
- [16] Minhas, N. M., Börstler, J., Petersen, K. (2023). Checklists to support decision-making in regression testing. *Journal of Systems and Software*, 202, 1-15. <https://doi.org/10.1016/j.jss.2023.111697>
- [17] Jeon, T., von Mayrhauser, A. (2002). *A knowledge-based approach to regression testing*. IEEE 1st Asia-Pacific Software Engineering Conference. Tokyo, Japan. <https://doi.org/10.1109/APSEC.1994.465265>
- [18] Marenbach, R., Albert, M. (2018). Regression test approach for testing of protection IEDs to improve field testing quality and support knowledge management. *The Journal of Engineering*, 2018 (15), 1023-1026. <https://doi.org/10.1049/joe.2018.0163>
- [19] Wnuk, K., Garrepalli, T. (2018). Knowledge management in software testing: A systematic snowball literature review. *e-Informatica Software Engineering Journal*, 12 (1), 51–78. <https://doi.org/10.5277/e-Inf180103>
- [20] de Souza, É. F., de Almeida Falbo, R., Vijaykumar, N. L., Felizardo, K. R., Meinerz, G. V., Specimille, M. S., Coelho, A. G. (2021). Development of an ontology-based approach for knowledge management in software testing: An experience report. *Journal of Software Engineering Research and Development*, 9 (1). 1-21. <https://doi.org/10.5753/jserd.2021.1898>
- [21] Chen, C. Y., Lee, J. C. (2022). Comparative effects of knowledge-based antecedents in different realms of CMMI-based software process improvement success. *Computer Standards & Interfaces*, 81. <https://doi.org/10.1016/j.csi.2021.103599>
- [22] Diepenbeck, M., Drechsler, R. (2015). Behavior driven development for tests and verification. En R. Drechsler, U. Kühne, *Formal Modeling and Verification of Cyber-Physical Systems* (pp, 275–277). Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-09994-7_11
- [23] Creswell, J. W. (2018). *A concise introduction to mixed methods research* (2nd Ed.). Sage Publications, Inc.