



UNIVERSIDAD VERACRUZANA
FACULTAD DE ESTADÍSTICA E
INFORMÁTICA

PROTOCOLO DE LA RSL:
“FACILIDAD DE EVOLUCIÓN (EVOLVABILITY) DEL
SOFTWARE”

MODALIDAD:
MONOGRAFÍA

PRESENTA:
MARIO EDUARDO DORANTES HERNÁNDEZ

DIRECTOR:
DRA. MARÍA KAREN CORTÉS VERDÍN

CO DIRECTOR:
MTRA. MARÍA DE LOS ÁNGELES ARENAS VALDÉS

XALAPA, VER. 2023

1 Necesidad de la revisión

Vivimos en un mundo de constante cambio, donde todo el tiempo se deben buscar alternativas para mejorar productos, servicios y procesos, entre otras cosas. En el contexto del software es aún más notoria esta necesidad de mejora, debido a los requisitos cambiantes por parte del mercado y del entorno. Dada esta situación, los sistemas de software deben modificarse constantemente, ya sea para añadir nuevas funcionalidades o para cambios en la tecnología. Por tal motivo, se busca que un sistema de software tenga la capacidad de evolucionar, ya que, gracias a esta característica, el sistema podrá adaptarse a cambios en el futuro sin sufrir demasiadas repercusiones. Es ideal considerar esta característica desde el inicio del desarrollo de un producto de software, para evitar estas repercusiones.

Este trabajo se centra en la búsqueda de información acerca de la facilidad de evolución. Se pretende investigar y analizar las diferentes propuestas y enfoques para medir la facilidad de evolución en el software, así como conocer los enfoques existentes para el diseño de software fácil de evolucionar.

2 Preguntas de investigación

A continuación, se muestran las preguntas de investigación, así como sus motivaciones principales, dichas preguntas son fundamentales para obtener los términos de búsqueda de este trabajo.

Pregunta de investigación	Motivación principal
RQ1- ¿Cuáles son las propuestas para definir la facilidad de evolución o facilidad de mantenimiento?	Conocer los enfoques, propuestas que a lo largo del tiempo han existido para definir la facilidad de evolución o facilidad de mantenimiento en el contexto del software
RQ2- ¿Cuáles son los elementos o aspectos que contribuyen a la facilidad de evolución o facilidad de mantenimiento?	Saber que elementos intervienen o definen la facilidad de evolución o facilidad de mantenimiento
RQ3- ¿Cómo se mide software fácil de evolucionar o fácil de mantener?	Saber si existen métricas para medir la facilidad de evolución o facilidad de mantenimiento en el software
RQ4- ¿Cómo se diseña software fácil de evolucionar o fácil de mantener?	Saber si existen estrategias y/o patrones de diseño que consideren la facilidad de evolución o facilidad de mantenimiento
RQ5- ¿Cuáles han sido las experiencias o resultados obtenidos al considerar la	Saber si ha habido experiencias utilizando algún método o estrategia

facilidad de evolución o facilidad de mantenimiento en el desarrollo de software?	para la facilidad de evolución o facilidad de mantenimiento en el desarrollo de software.
---	---

3 Estrategia de búsqueda

En la presente sección, se muestra la estrategia de búsqueda, la cual consta de los términos de búsqueda, la cadena de búsqueda resultante de dichos términos y las fuentes donde se realizará la investigación de los estudios, los cuales serán una pieza fundamental para el desarrollo de este trabajo.

3.1 Términos de búsqueda

A continuación, se muestran los términos relacionados con el tema, que se obtienen tomando como base las preguntas de investigación. Dichos términos son un predecesor de la cadena de búsqueda.

Palabra clave	Término en inglés	Sinónimo o palabra relacionada	Término(s) de búsqueda
Facilidad de evolución	Evolvability		Evolvability
Facilidad de mantenimiento	Maintainability	Software Maintainability	Software Maintainability
Medición	Measurement	Metric	Measurement Measure Metric
Diseño de software	Software design	Software Design	Software design
Patrón	Pattern	Method Process Strategy Framework	Pattern Design pattern Architectural pattern Method Process Strategy Framework

3.2 Selección de fuentes

Para la selección de fuentes, se tomó en cuenta fuentes de las cuales la Universidad Veracruzana tiene acceso.

- IEEE Xplore
- ACM Digital Library
- Springer
- Elsevier

3.3 Cadena de búsqueda

A continuación, se presenta la cadena de búsqueda creada a partir de los términos de búsqueda obtenidos anteriormente.

(evolvability OR "software maintainability") AND (measurement OR measure OR metric OR "software design" OR pattern OR "design pattern" OR "architectural pattern" OR method OR process OR strategy OR framework) AND NOT (neural OR robotic* OR geneti* OR mechanical OR artificial OR evolutionary)

4 Selección de estudios primarios

En la presente sección, se muestran los criterios de selección de los estudios primarios, estos son los criterios de inclusión, así como los de exclusión. De igual manera, se presenta el procedimiento a seguir para la selección de dichos estudios.

4.1 Criterios de selección de estudios primarios

Criterios de inclusión:

- CI-1. Se considerarán solamente estudios primarios en inglés.
- CI-2. El título debe hacer referencia a desarrollo de software.
- CI-3. El título y/o abstract debe contener el término de búsqueda "Evolvability", "Evolution" o "Maintainability"
- CI-4. Después de leer el abstract, se muestran indicios que el estudio contesta al menos una pregunta de investigación.

Criterios de exclusión:

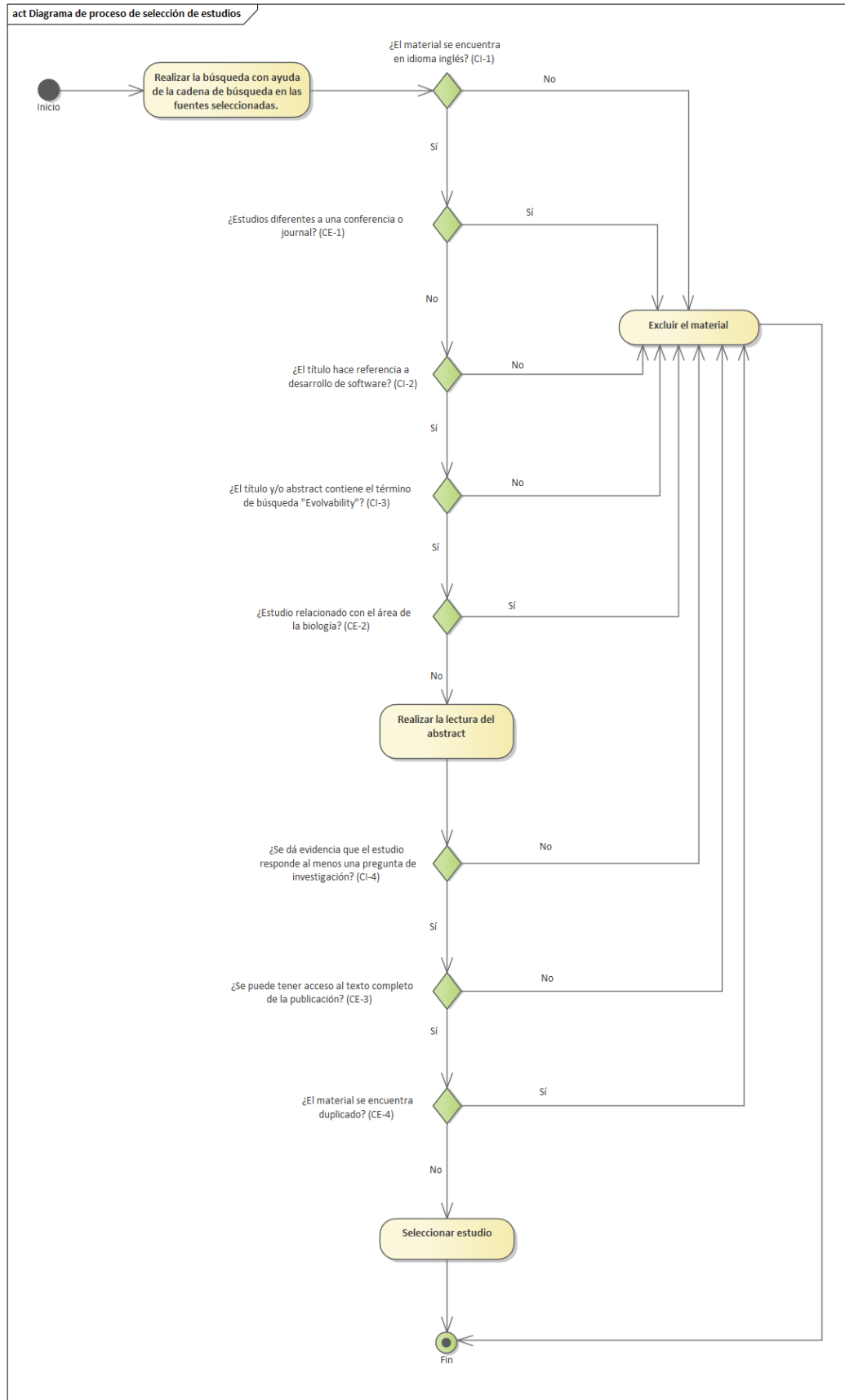
- CE-1. Estudios primarios diferentes a una conferencia o journal.
- CE-2. No se considerarán estudios relacionados con el área de la biología.
- CE-3. No es posible obtener acceso al material completo.
- CE-4. Se trata de un estudio duplicado.

4.2 Procedimiento de selección de estudios primarios

1. Realizar la búsqueda con ayuda de la cadena de búsqueda en las fuentes seleccionadas.
2. Considerar solamente estudios primarios en inglés. (CI-1)
3. Excluir estudios que sean diferentes a una conferencia o journal. (CE-1)
4. Verificar que el título hace referencia a desarrollo de software. (CI-2)
5. Verifica que el título y/o abstract contiene el término de búsqueda “Evolvability”, “Evolution” o “Maintainability” (CI-3)
6. Excluir estudios relacionados con el área de la biología. (CE-2)
7. Realizar la lectura del abstract y verificar que se dé evidencia que el estudio responde al menos una pregunta de investigación. (CI-4)
8. Revisar que se pueda tener acceso al texto completo de la publicación (CE-3)
9. Revisar que el material no este duplicado. (CE-4)

A continuación, se presenta un diagrama de actividad que muestra el proceso de selección de estudios primarios.

act Diagrama de proceso de selección de estudios



5 Estrategia de extracción de información

A continuación, se muestra la estrategia para la extracción de información de los estudios que cumplan con los criterios previamente mencionados.

Datos del estudio					
Título	Título del artículo				
Autores	Los nombres de los autores				
Año	Año del artículo				
Fuente	Nombre de la base de datos				
Tipo de publicación	Clasificación del artículo				
Referencia o DOI	DOI del artículo o número de identificador del artículo				
Palabras clave	Términos clave del artículo				
Abstract o resumen	Abstract o resumen del artículo				
Pregunta(s) de investigación relacionada(s)	RQ1	RQ2	RQ3	RQ4	RQ5
	Se marca la casilla en caso de relacionarse con la pregunta	Se marca la casilla en caso de relacionarse con la pregunta	Se marca la casilla en caso de relacionarse con la pregunta	Se marca la casilla en caso de relacionarse con la pregunta	Se marca la casilla en caso de relacionarse con la pregunta
Extracción para la síntesis					
RQ1- ¿Cuáles son las propuestas para definir la facilidad de evolución o facilidad de mantenimiento?	Propuestas	Información extraída relacionada con las propuestas para definir la facilidad de evolución o facilidad de mantenimiento			
	Enfoques	Información extraída relacionada con los enfoques para definir la facilidad de evolución o facilidad de mantenimiento			
RQ2- ¿Cómo se diseña software fácil de evolucionar o fácil de mantener?	Estrategias	Información extraída relacionada con las estrategias de diseño que consideren la facilidad de evolución o facilidad de mantenimiento			
	Patrones	Información extraída relacionada con los patrones de diseño que consideren la facilidad de evolución o facilidad de mantenimiento			

	Métodos	Información extraída relacionada con los métodos de diseño que consideren la facilidad de evolución o facilidad de mantenimiento
	Procesos	Información extraída relacionada con los procesos de diseño que consideren la facilidad de evolución o facilidad de mantenimiento
	Frameworks	Información extraída relacionada con los Frameworks de diseño que consideren la facilidad de evolución o facilidad de mantenimiento
RQ3- ¿Cómo se mide software fácil de evolucionar o fácil de mantener?	Métricas	Información extraída relacionada con las métricas para medir la facilidad de evolución o facilidad de mantenimiento
RQ4- ¿Cuáles son los elementos o aspectos que contribuyen a la facilidad de evolución o facilidad de mantenimiento?	Elementos	Información extraída relacionada con los elementos que intervienen en la facilidad de evolución o facilidad de mantenimiento
	Aspectos	Información extraída relacionada con los aspectos que intervienen en la facilidad de evolución o facilidad de mantenimiento
RQ5- ¿Cuáles han sido las experiencias o resultados obtenidos al considerar la facilidad de evolución o facilidad de mantenimiento en el desarrollo de software?	Experiencias	Información extraída relacionada con las experiencias obtenidas al considerar la facilidad de evolución o facilidad de mantenimiento
	Resultados	Información extraída relacionada con los resultados obtenidos al considerar la facilidad de evolución o facilidad de mantenimiento

6 Estrategia para la síntesis de datos

Para presentar los resultados obtenidos de este trabajo, se seleccionó el tipo de síntesis narrativa. Debido a que este tipo de síntesis nos permite resumir y explicar de manera textual los hallazgos encontrados dentro de la investigación, al igual que,

los datos se tabulan de una manera que es consistente con las preguntas de investigación planteadas.

7 Limitaciones

A continuación, se presentan las limitaciones de este trabajo.

- Únicamente una RSL sobre la facilidad de evolución para conocer el estado del arte
- Solo se cuenta con el tiempo del periodo escolar para realizar la RSL
- Desconocimiento del tema a tratar
- Información limitada a solo las bases de datos que a través de CONRICyT la Universidad Veracruzana proporciona

8 Informe

Se estableció el siguiente índice para el reporte siguiendo la guía de (Kitchenham, Budgen & Brereton. 2015).

- Título
- Resumen
 - Contexto
 - Objetivos
 - Métodos
 - Resultados
 - Conclusiones
- Palabras clave
- Contexto o introducción
 - Preguntas de investigación
- Protocolo de revisión
 - Criterios de inclusión y exclusión
 - Proceso de búsqueda
 - Evaluación de la calidad
 - Extracción y síntesis de datos
- Resultados
- Amenazas a la validez
- Conclusiones
- Agradecimientos
- Referencias

9 Gestión de la revisión

Para la gestión de este trabajo, se planeó el siguiente cronograma de actividades. Además, para administrar las referencias se eligió la herramienta Mendeley.

Cronograma de actividades

Actividades	Septiembre				Octubre				Noviembre				Diciembre			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Desarrollar de primera versión del protocolo																
Preparar la primera presentación																
Realizar la primera presentación																
Finalizar el protocolo de la RSL																
Realizar el proceso de búsqueda																
Preparar la segunda presentación																
Realizar la segunda presentación																
Seleccionar estudios																
Realizar el proceso de evaluación																
Realizar el proceso de extracción de datos																
Sintetizar la información																
Redactar reporte																
Preparar la tercera presentación																
Tercera presentación																

10 Referencias

Kitchenham, B. A., Budgen, D., & Brereton, P. (2015). *Evidence-based software engineering and systematic reviews*.