|  |  |
| --- | --- |
| Nº línea |  |
| 1 | **Software Release Patterns: When is it a good time to update a software** |
| 2 | **component**? |
| 3 | **Abstract** |
| 4 | Over the past decade the industry 4.0 witnessed a trend towards an increasing |
| 5 | number of software components, dependencies towards third party software |
| 6 | components, and software component release cycles. Industry 4.0 teams building |
| 7 | software products. are more frequently impacted by third party software component |
| 8 | updates. Due to this dependency, updating a single third party software component |
| 9 | can break an entire software product. Reasons include parallel conflicting updates of |
| 10 | third party software components, updating to an unstable version, or updating to a |
| 11 | major stable version without an impact analysis. The objective of this paper is to |
| 12  13 | reduce the risk of breaking updates by reviewing software release patterns and  proposing update scheduling recommendations. |
| 14 | **Abstract** |
| 16 | **Keywords :**Software Updates, Software Dependencies, Software Release Schedule. |
|  |  |

**Referencia:**

Solomon Berhe, Marc Maynard & Foutse Khomh.(2020). Software Release Patterns: when is it a good time to update a software component*? .The 3rd International Conference on Emerging Data and Industry.4.0(EDI40).*<http://creativecommons.org/licenses/by-nc-nd/4.0/>

* **ESTUDIANTE: JOSE AMANDO MORA RIVAS 29.634.375**

**SEM12 04JoseMora**

Reconozco que en la última década los componentes del software, han sufrido numerosos lanzamientos de productos los cuales afectan a estos con una frecuencia, debido a las constantes actualizaciones.Resaltada de la Linea 4 a la 8. respectivamente / Over the past decade the industry 4.0 witnessed a trend towards an increasing number of software components, dependencies towards third party software components, and software component release cycles. Industry 4.0 teams building software products. are more frequently impacted by third party software component updates. /Durante la última década, la industria 4.0 fue testigo de una tendencia hacia un aumento número de componentes de software, dependencias hacia software de terceros componentes y ciclos de lanzamiento de componentes de software. Creación de equipos de Industria 4.0 productos de software. Se ven afectados con más frecuencia por componentes de software de terceras actualizaciones./ que resalte con este color.

Se observa, una comparacion cuando se presenta un software estable e inestable ; debido a la dependecia de sus diverzas modificadificacion o actualizaciones que pueden llevar al rompimimeto de los componestes de dicho software, afectado su funcionamiento .Esto va en las lineas resaltadas de la 8 a la 11 /Due to this dependency, updating a single third party software component can break an entire software product. Reasons include parallel conflicting updates of third party software components, updating to an unstable version, or updating to a major stable version without an impact analysis./ Debido a esta dependencia, la actualización de un solo componente de software de terceros puede romper un producto de software completo. Las razones incluyen actualizaciones contradictorias paralelas de componentes de software de terceros, actualización a una versión inestable o actualización a una versión principal estable sin un análisis de impacto./que resalte con este color

En este caso se denota, que el objetvo de éste fragmento ,es que se reduzca el riesgo y prevenga la interrupcion aquel patrones de lanzamiento de software , dando tambien una serie de recomendaciones y propuestas referetes a dichas actualizaciones de programas y funcionamiento. Esto ubicado en la linea 11 a la 13 /The objective of this paper is to reduce the risk of breaking updates by reviewing software release patterns and proposing update scheduling recommendations /. El objetivo de este trabajo es que reduzca el riesgo de interrumpir las actualizaciones revisando los patrones de lanzamiento de software y proponiendo recomendaciones de programación de actualizaciones./ que resalte con este color.

Mis comentarios:

- Realizaste un muy buen trabajo de análisis y comprensión del resumen del AIC 1 que seleccionaste.

- José Amado, te pregunto: ¿Qué medio/recurso/tecnología usaste para la traducción de cada una de las partes que resaltaste en el resumen? Espero tu respuesta.

- Sabemos que el resumen sigue la estructura básica IMRyD que tienen los AIC. En tu resumen, ¿dónde se encuentran estos elementos; es decir, la introducción, la metodología, los resultados y/o la discusión y conclusión? Respondo esta pregunta de la siguiente manera:

La **introducción** se observa entre las líneas 4 a 8 (*Over*... *updates*) porque es un comentario general que introduce el tema de estudio (*Over the past decade*...). La **metodología** se encuentra implícita (sobreentendida) entre las líneas 8 a 11 (*Due to*... *impact analysis*) porque dejan ver cómo se desarrolla el estudio (*updating*... [actualizando]). No hay referencia al **resultado**. Luego, se evidencia el **objetivo** (*the objective of this paper is*), el cual forma parte de la introducción y debe aparecer con claridad en todo resumen. Aunque no esté cerca de la introducción (como se observa aquí), es fundamental que el objetivo aparezca y se plantee con claridad.

- La estructura IMRyD de un AIC también está presente en el resumen. Sin embargo, una conclusión interesante de esta tarea es que esa estructura se encuentra *en teoría* en tu resumen; es decir, aquí comprobamos que no es así necesariamente (algo así como que “toda regla tiene su excepción”).

- Relaciona estos comentarios con la estructura de tu AIC 2.

- Calificación: 18 puntos.