

Actividad 6 - Taller técnica de mantenimiento de software

Mantenimiento de Software

Heiner Duvan Acosta Urrego

hacosta@ibero.edu.co

ID: 100086805

1. INTRODUCCION

La comprensión de software es una técnica de el mantenimiento de software la cual es primordial para este proceso ya que si no comprendemos el software, no podemos interpretarlo y mucho menos corregirlo ni adecuarlo, en este documento encontraremos la definición de comprensión de software y su aplicación .

2. Compresión del Software

La comprensión del software como su nombre lo indica se basa en comprender el software y así llegar a entender que cambios, modificaciones o mejoras que debemos hacer, existen sistemas de compresión de software que son herramientas claves para lograr comprender el software otra forma de compresión del software y la acertada es una buena documentación. La comprensión de un software puede tener diferentes significados o perspectivas. Podemos estar interesados en la forma en que la computadora ejecuta las instrucciones con el objetivo de entender el flujo de control; o podemos enfocar nuestro interés en los efectos que producirá la ejecución del programa en el dispositivo (físico o virtual) que es manipulado por el mismo. En cualquiera de los dos casos pensamos que una herramienta de inspección visual es fundamental para efectuar este tipo de tareas.

3. Plan de ejecución de la técnica seleccionada

- Recibir toda la información de quien entrega el software.
- Indagar respecto a la construcción y funcionalidad de este software.
- Tomar la documentación del software y leerla minuciosamente.
- Utilizar una herramienta para inspeccionar y comprender el software podría ser SHriMP si el código esta escrito en java
- Ver el código fuente.
- Realizar un análisis y pruebas de ejecución sobre este software con respecto a todo lo encontrado.
- Implementar las mejoras y/o actualizaciones pertinentes en este software.

4. Comprensión de software sobre JUNIORAPP

Podemos realizar los pasos mencionados anteriormente para realizar la comprensión del software **JUNIORAPP**, encontramos que es un software que en el momento se encuentra en desarrollo por tal motivo hay muchas mejoras que realizarle , se encontró que está hecho en java y MySQL, por lo que lo hace un software robusto se encontró que le hacen falta varias clases para llegar al objetivo que se busca respecto a lo encontrado a su documentación, podemos seguir utilizando este código para poder implementar mejoras y actualizaciones sobre el mismo. Podemos pensar en una migración a lenguajes menos robustos por el consumo de recursos en la web facilite a los clientes que la utilicen.

5. Conclusiones

- Definitivamente la comprensión de software es primordial para las técnicas del mantenimiento de software.
- Esta técnica la mayoría de programadores la utilizan.
- Las demás técnicas utilizan esta técnica directa o indirectamente.
- Comprendiendo el software en su totalidad podemos asegurarnos de que obtenga las mejoras requeridas.
- Existen varias herramientas para la comprensión de software que nos pueden facilitar la vida.

Enlace a [GITHUB](#)

6. Referencias

- Bourque, P. & Fairley, R. E. (Eds). (2014). Swebok. Guide to the Software Engineering Body of Knowledge - Versión 3.0. IEEE Computer Society. Recuperado de <https://ieeecs-media.computer.org/media/education/swebok/swebok-v3.pdf>