**Autoevaluación UT3.-Optimización y Documentación.**

1. La refactorización:
   1. Se utiliza como técnica complementaria de realización de pruebas.
   2. Genera un código más difícil de entender y de mantener.
   3. Utiliza una serie de patrones, de aplicación sobre el código fuente.
   4. Es una técnica de programación no recogida por los entornos de desarrollo.
2. ¿Cuál no es un patrón de refactorización?
   1. Eliminar parámetros de un método.
   2. Renombrado.
   3. Sustitución de un bloque de sentencias por un método.
   4. Mover clase.
3. Relaciona los componentes de un sistema de control de versiones con su definición:

|  |  |  |
| --- | --- | --- |
| Componente |  | Definición |
| Repositorio |  | En un directorio específico del repositorio. Puede identificar una parte del proyecto o ser el proyecto por completo. |
| Módulo |  | Información textual que se añada a un conjunto de archivos o a un módulo completo para indicar alguna información importante. |
| Revisión |  | Es el lugar de almacenamiento de los datos de los proyectos. Suele ser un directorio en algún ordenador. |
| Etiqueta |  | Revisiones paralelas de un módulo para efectuar cambios sin tocar la evolución principal. Se suele emplear para pruebas o para mantener los cambios en versiones antiguas. |
| Rama |  | Es cada una de las versiones parciales o cambios en los archivos o repositorio completo. La evolución del sistema se mide en revisiones. Cada cambio se considera incremental |

1. Relaciona las órdenes que se pueden realizar en un sistema de control de versiones con su acción.

|  |  |  |
| --- | --- | --- |
| Órden |  | Acción |
| Checkout |  | Actualiza la copia con cambios recientes en el repositorio. |
| Update |  | Almacena la copia modificada en el repositorio. |
| Commit |  | Abandona los cambios en la copia de trabajo |
| Abort |  | Obtiene una copia del trabajo para poder trabajar con ella |

1. ¿Qué afirmación sobre control de versiones es correcta?
   1. Solo puede existir una única versión de una clase.
   2. El almacenamiento de versiones es local a cada máquina.
   3. El repositorio centraliza el almacenamiento de los datos.
   4. Ninguna de las anteriores es correcta.
2. ¿Qué herramienta **no es** una herramienta de Control de Versiones?
   1. Subversion.
   2. CVS.
   3. Mercurial.
   4. PMD
3. ¿Cuál de las siguientes no es una tarea básica de la Gestión de Configuraciones del Software (GCS)?
   1. Control de cambios.
   2. Generación de informes.
   3. Auditorías de configuraciones.
   4. Gestión del repositorio
4. La Gestión de Configuraciones de Software (GCS):
   1. Verifica la evolución de la implementación de una clase.
   2. Es una tarea que se puede gestionar con herramientas de Control de Versiones.
   3. Es un conjunto de actividades desarrolladas para gestionar los cambios a lo largo del ciclo de vida del software .
   4. Ninguna de las anteriores.
5. Un comentario en formato JavaDoc.
   1. Utiliza los caracteres //
   2. Comienzan con /\* y termina por \*/
   3. Comienza por /\*\* y terminan por \*/
   4. Ninguna de las anteriores.
6. Relaciona los tags más habituales de **Javadoc** con su descripción:

|  |  |  |
| --- | --- | --- |
| TAG |  | Descripción |
| @version. |  | Nombre del autor. |
| @author. |  | Parámetros que recibe el método. |
| @since. |  | Comentario sobre las excepciones que lanza. |
| @param |  | Comentario con datos indicativos del número de versión |
| @return. |  | Significado del dato devuelto por el método |
| @throws. |  | Indicación de que el método es obsoleto. |
| @deprecated. |  | Fecha desde la que está presente la clase. |

1. Los sistemas de control de versiones :
   1. Son distribuidos.
   2. Son centralizados.
   3. Pueden ser distribuidos y centralizados.
   4. Ninguna de las anteriores.
2. Que cada usuario tenga su repositorio local corresponde a un sistema de control de versiones:
   1. Centralizado.
   2. Distribuido.
   3. Ninguna de las anteriores.
3. Git es un software de control de versiones:
   1. Distribuido.
   2. Centralizado.
   3. Ninguna de las anteriores.
4. Github es:
   1. Un sistema de control de versiones.
   2. Es una plataforma de desarrollo colaborativo que utiliza Git como SCV.
   3. Ninguna de las anteriores.
5. Selecciona la afirmación correcta respecto a las siguientes herramientas de sistemas de control de versiones (SCV)
   1. Subversión (SVN) es centralizada.
   2. Mercurial es centralizada.
   3. Microsoft Visual Source Safe es distribuida.
   4. Ninguna de las anteriores es correcta.
6. Los Elementos de un Sistema de Configuración (ECS) pueden ser:
   1. Programas.
   2. Documentos.
   3. Datos.
   4. Todos los anteriores.
   5. Ninguna respuesta es correcta.
7. Entre las actividades de un sistema de configuración, tenemos:
   1. Gestión del cambio.
   2. Control de versiones.
   3. Las dos anteriores.
   4. Ninguna respuesta es correcta.
8. Bugzilla es una herramienta CASE para:
   1. El control de versiones.
   2. La Gestión del cambio.
   3. Las dos anteriores.
   4. Ninguna de las anteriores es correcta.
9. La documentación generada automáticament con javadoc:
   1. Se almacena en una carpeta “dist” dentro del proyecto.
   2. Se genera un fichero .pdf
   3. Ninguna de las anteriores.
10. PMD es:
    1. Una herramienta de control de versiones.
    2. Una analizador de código.
    3. Una herramienta para la gestión del cambio.
    4. Ninguna de las anteriores.