**ENTORNOS DE DESARROLLO - 1º DAM**

**UD1. ELEMENTOS DEL DESARROLLO DE SOFTWARE**

E03 - PREGUNTAS SOBRE EL APARTADO 4

|  |  |
| --- | --- |
| **Resultado de Aprendizaje 1**: Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento. | |
| **Criterio de evaluación** | **Ejercicio/s** |
| RA01.b Se han clasificado los lenguajes de programación, identificando las principales características de Java. | TODOS |

* ¿Cómo se llama el creador de Java?

James Gosling

* ¿A partir de qué lenguaje está creado Java?

C++

* ¿En qué fecha fue lanzado Java?

En el otoño de 1995

* Enumera las razones que motivaron la creación de Java.

Creciente necesidad de interfaces mucho más cómodas, fiabilidaddel código y facilidad de desarrollo y enorme diversidad de controladores electrónicos.

* Enumera las fechas y el número de versión de Java desde la versión 1.0, liberada en el año 1995.

En diciembre de 1998, Sun anuncia el Java 2 como marca comercial de la versión JDK 1.2. Sun, la empresa que desarrollaba Java fue adquirida por Oracle en 2010, en septiembre de 2017 se libera la versión 9 de Java,

* Enumera las características de Java.

Lenguaje simple, orientado a objetos, distribuido, compilado e interpretado, robusto, portable, dinámico, multihilo, seguro, independiente a la arquitectura.

* ¿Por qué Java es simple?

Java posee una curva de aprendizaje muy rápida. Resulta relativamente sencillo

escribir programas interesantes desde el principio. Todos aquellos familiarizados con C++ encontrarán que Java es más sencillo, ya que se han eliminado ciertas características, como los punteros. Debido a su semejanza con C y C++, y dado que la mayoría de la gente los conoce aunque sea de forma elemental,resulta muy fácil aprender Java. Los programadores experimentados en C++ pueden migrar

muy rápidamente a Java y ser productivos en poco tiempo.

¿Qué significa que Java es independiente de la arquitectura?

Java está diseñado para soportar aplicaciones que serán ejecutadas

en los más variados entornos de red, desde Unix a Windows, pasando por Mac y estaciones de trabajo,sobre arquitecturas distintas y con sistemas operativos diversos. Para acomodarrequisitos de ejecucióntan variopintos, el compilador de Java genera bytecodes: un formato intermedio indiferente a la arquitectura, diseñado para transportar el código eficientemente a múltiples plataformas hardware y software. Elresto de problemas los soluciona el intérprete de Java

¿Y portable?La indiferencia a la arquitectura representa sólo una parte de su portabilidad. Además, Java especifica los tamaños de sus tipos de datos básicos y el comportamiento de sus operadores aritméticos, de manera que los programas son iguales en todas las plataformas. Estas dos últimas características se conocen como la Máquina Virtual Java (JVM)

* ¿Por qué Java es compilado e interpretado?Java es compilado, en la medida en que su código fuente se transforma en una especie de código máquina, los bytecodes, semejantes a las instrucciones de ensamblador. Por otra

parte, es interpretado, ya que los bytecodes se pueden ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real (runtime)

* ¿Qué es y para qué sirve la máquina virtual de Java (JVM)?

Esta “máquina” es un software necesario para poder ejecutar cualquieraplicación escrita en java y es el encargado de traducir el bytecode en lenguaje máquina, ya adaptado a la máquina concreta sobre la que se está ejecutando. Portanto, será necesario instalarla JVM apropiada al S.O. del dispositivo.

**Ejercicios 04 y 05 de Entorno de desarrollo**

\*¿Que es una metodología de desarrollo de software? ¿Para qué sirve?

Es la acción que lleva a cabo un programador para crear o desarrollar software. Estos especialistas en informática conciben y elaboran sistemas informáticos, los implementan y los ponen en marcha para ser utilizados con uno o varios lenguajes de programación. Sus principales características son: la programación orientada al objeto y la separación de las distintas etapas lógicas en función de la aplicación, el acceso y el nivel de presentación.

El desarrollo de software está impulsando la creación de aplicaciones corporativas, tanto a nivel interno como externo, y es un proceso complejo que requiere mucha planificación. Sin embargo, la creación de aplicaciones no implica simplemente su desarrollo, sino también otras tareas indispensables para su funcionamiento, como: el análisis de los sistemas, el diseño del software, la prueba y la revisión, el mantenimiento, la usabilidad, la reingeniería, la arquitectura, entre muchas otras cosas.

Los principales usos del desarrollo de software en la actualidad están llevándose a cabo por parte de las empresas, quienes están creando sus propias aplicaciones tanto móviles como web para mejorar el proceso interno, la atención al cliente, la visibilidad de la marca y la experiencia de usuario. Las aplicaciones móviles son una vía perfecta para dar a conocer tus productos y servicios y llegar de manera más rápida, cómoda y sencilla a los consumidores.

\*Enumera ordenadamente, las fases que componen un proyecto de desarrollo de software.

1 Análisis de sistemas y requisitos, 2 Diseño y arquitectura de software, 3 Programación e implementación, 4 pruebas y revisión, 5 Mantenimiento y cuidado,

6 Documentación, 7 Diseño de usabilidad.

\*¿Para qué sirve la fase de pruebas?

Sirven para detectar las disconformidades de los trabajadores, los clientes y los errores. Es el momento de comprobar que el software funciona correctamente con las tareas indicadas, y para ello se pueden realizar las pruebas de cada módulo por separado y después de forma integral. Las revisiones, por su parte, son la oportunidad perfecta para ejecutar desde el principio del desarrollo las medidas necesarias para asegurar la calidad del sistema.

\*¿Que es una metodología àgil?¿Que la diferencia de cualquier otra metodología?

En concreto, las metodologías ágiles de desarrollo de software buscan proporcionar en poco tiempo pequeñas piezas de software en funcionamiento para aumentar la satisfacción del cliente. Estas metodologías utilizan enfoques flexibles y el trabajo en equipo para ofrecer mejoras constantes.

Los sistemas tradicionales se centran en la planificación proactiva donde factores como el costo, el alcance y el tiempo son importantes, pero la gestión ágil de proyectos prioriza el trabajo en equipo, la colaboración con los clientes y la

flexibilidad.

\*¿Cuáles son los roles más importantes en scrum?

Está conformado por 3 roles principales: El Product Owner (Dueño del Producto), el Scrum Master (Dueño del proceso) y el Team (Miembros del Equipo de Desarrollo)

\*¿Cuáles son los pasos a seguir en una interación o sprint?

1 Planificación del sprint, 2 Etapa de desarrollo, 3 Revisión del sprint, 4 Retroalimentación

\*¿Qué es el Product Backlog o pila de producto?

El Product Backlog o pila de producto en un proyecto que sigue la metodología Scrum consiste en una lista con todos los requerimientos iniciales del producto que se va a desarrollar. Se trata de una lista dinámica, que irá evolucionando a medida que lo hace el producto y el entorno del proyecto.

\*¿Cuál debe ser la duración aproximada de una interacción o sprint?

Su duración debe ser de 30 días o menos. Normalmente su duración se determina en semanas (mínimo 1 semana, y máximo 4 semanas)