

Pauta de evaluación Sprint Módulo 2:

“Fundamentos de programación en Java”

Módulo 2	Puntaje máximo	Puntaje mínimo de aprobación	Escala de evaluación
Sprint 2	70 pts	49 pts	70%

REGLAS E INSTRUCCIONES GENERALES:

- Lea atentamente el caso de negocio y comprenda tanto su modelo como sus reglas.
- La evaluación es de carácter grupal, si usted desea trabajar solo, **debe avisar a su tutor principal o en su defecto al tutor adviser antes de subir su evaluación a plataforma.**
- Al momento de entregar su evaluación, **“en la documentación de autores del software”** debe incluir a todos los miembros de su equipo, **en caso contrario, se le descontarán 3 pts.** Además, todos los integrantes del grupo deben subir su prueba a plataforma.

CONTENIDOS A EVALUAR:

Se evaluarán los conocimientos adquiridos durante los aprendizajes esperados 3 y 4. A continuación, se deja a disposición un listado de los contenidos que serán evaluados durante el sprint.

Aprendizaje esperado 3:

- Correcto uso del entorno de programación Java.
- Correcto uso de variables y sus tipos, así como también el uso adecuado de operadores lógicos.
- Correcto uso de ciclos y validaciones, según operaciones lógicas.
- Correcto uso de arreglos, colecciones y estándares y/o convenciones de programación.

Aprendizaje esperado 4:

- Correcto uso de Api Java, uso de documentación JavaDoc, uso de breakpoints durante el Debugging.
- Creación de clases, objetos, métodos y constructores basados en principios de programación orientada a objetos.
- Correcto uso de modificadores de acceso, getters and setters y colaboración entre objetos.
- Diseño óptimo de diagramas de clase, en base a los estándares y convenciones de la Programación orientada a objetos.
- Creación de herencias, interfaces, aplicación de polimorfismo en base a principios de la POO.
- Aplicación de principios SOLID.

Nota: Tanto para el criterio de evaluación **“Api - Java”** como el de **“diagramas de clases”** se solicita subir evidencias fotográficas. Se sugiere crear una carpeta llamada **“Evidencias”(dentro del directorio raíz de su app)** que contenga todas las capturas de pantalla requeridas para los criterios de evaluación mencionados anteriormente.

Criterio	Excelente 7	Bueno 5	Regular 3	suficiente 1	No cumple 0
APRENDIZAJE ESPERADO 3					
Sintaxis y compilación	El programa no presenta errores de sintaxis y compila sin problemas.	El programa compila sin problemas, pero presenta algunas falencias en su sintaxis.	El programa compila, pero presenta 1 error en su sintaxis.	El software compila, pero presenta múltiples errores en su sintaxis.	El software no compila
Uso de Funciones Integer y/o String	El software hace uso de funciones integer y/o string en todo el código según corresponda.	El software presenta entre un 80% y un 90% de uso de funciones Integer y/o string, según corresponda.	El software presenta un 40% hasta un 70% de uso de funciones Integer y/o string, según corresponda.	El software no presenta un 50% de uso de funciones Integer y/o String, según corresponda.	El software no presenta uso de funciones Integer y/o String
Validaciones de campos obligatorios.	El software respeta en su totalidad las validaciones de campos obligatorios.	El software respeta entre un 80% y un 90% de las validaciones de campos obligatorios.	El software respeta entre un 60% y un 70% de las validaciones	El software cumple con un 50% de las validaciones de campos obligatorios.	El software no valida los campos obligatorios.
Validaciones de formato y extensión de los campos solicitados.	El software cumple con la totalidad de los formatos y extensiones de los campos solicitados.	El software cumple entre un 80% y un 90% de los formatos y extensiones de los campos solicitados.	El software cumple entre un 60% y un 70% de los formatos y extensiones de campos solicitados.	El software cumple con un 50% de los formatos y extensiones de campos solicitados.	El software no respeta los formatos de los campos.

APRENDIZAJE ESPERADO 4

Complejidad de las clases	El software contiene todos los atributos, métodos, getters, setters(incluyen do de herencia, si se desean modificar) solicitados en el modelo del proceso, de esto se solicita <u>cubrir un 100% del total de las clases creadas.</u>	El software contiene todos los atributos, métodos, getters, setters(incluye ndo de herencia, si se desean modificar) solicitados en el modelo del proceso, de esto se solicita <u>cubrir entre un 80% y un 90% de las clases creadas.</u>	El software contiene todos los atributos, métodos, getters, setters (incluyendo de herencia, si se desean modificar) solicitados en el modelo del proceso, de esto se solicita <u>cubrir entre un 60% y un 70% del total de las clases creadas.</u>	El software contiene todos los atributos, métodos, getters, setters(incluy endo de herencia, si se desean modificar) solicitados en el modelo del proceso, de esto se solicita <u>cubrir un 50% del total de las clases creadas.</u>	El software contiene menos del 50% de las clases completadas.
Uso de API Java	<u>Implementa Javadoc en todo el desarrollo del software</u> y, además, se evidencia (subir capturas) el uso de al menos 5 breakpoints.	<u>Implementa Javadoc en todo el desarrollo del software</u> y, además, se evidencia (subir capturas) el uso de 3 breakpoints.	<u>Implementa Javadoc en todo el desarrollo del software,</u> incluyendo (comentarios de código, definición de clase, versión, autores, parámetros, polimorfismo, herencia, etc)	El software cuenta sólo con comentarios de código simple.	El software no contiene documentación Javadoc ni comentarios simples. Además el equipo no adjunta evidencia de breakpoints.
Uso de principios SOLID	Aplica y reconoce(MENCIONA EN CÓDIGO) los 5 principios SOLID	Aplica y reconoce 3 de 5 principios SOLID	Aplica y reconoce 2 de 5 principios SOLID	Aplica y reconoce 1 de 5 principios SOLID	El equipo no identifica ni aplica los principios SOLID

Uso de estándares convenciones y Estilo de codificación	Se aplicaron estándares, convenciones y estilos de codificación en todo el proyecto.	Se aplicaron estándares, convenciones y estilos de codificación entre un 80% y un 90% del proyecto	Se aplicaron estándares, convenciones y estilos de codificación entre un 70% y un 60% del proyecto	Se aplicaron estándares, convenciones y estilos de codificación en un 50% del proyecto.	No aplica ningún estándar, convención y/o estilos de codificación
Diagramas de Clases	El equipo entrega el diagrama de clases, este se encuentra completo y sin errores de diseño.	El equipo entrega el diagrama de clases, este se encuentra completo pero con errores de baja gravedad tales como, mala cardinalidad, asociaciones erróneas, métodos y atributos con mal encapsulación.	El equipo entrega el Diagrama de clases, se encuentra completo, pero contiene errores graves de diseño, herencias, interfaces, polimorfismos mal aplicados.	El equipo entrega el Diagrama de clases, pero está incompleto, faltan clases, asociaciones, herencias e interfaces, etc.	El equipo no entrega diagrama de clases