

Portafolio Educativo para el curso

**Desarrollo de Aplicaciones Móviles Android
Trainee**

Tabla de contenido

1. Guía de elementos que determina el tipo de trabajo	3
2. Estrategia didáctica	5
3. Introducción al Portafolio	6
4. Temas Centrales del Portafolio	8
5. Apartado de Cierre	10
6. Definición de plataforma digital o herramienta que se utilizará para publicación y trabajo del mismo	11
7. Instrumento evaluativo asociado al portafolio de proyecto	12

1. Guía de elementos que determina el tipo de trabajo

El siguiente documento describe la creación de un portafolio educativo utilizando una plataforma GitHub como parte de la estrategia didáctica para el desarrollo del módulo N°1

Cada estudiante, utilizando un computador y una conexión a internet, deberá crear una cuenta gratuita en la plataforma GitHub que se detalla más adelante.

Esto le permitirá a cada estudiante crear un repositorio remoto el cual deberá llamar “Módulo Programación Básica en Java”. Este repositorio será utilizado durante todo el módulo como portafolio educativo, a través del cual se llevará un registro de todo el proceso de aprendizaje.

Este repositorio contendrá diversos directorios en su interior. Los primeros elementos serán nuevos directorios, los cuales contendrán todos los archivos relacionados con las ejecuciones de cada sesión de clases. El segundo elemento será una página de texto llamado “Documentación”, el cual deberá contener el registro de cada ejecución realizada por el estudiante, acompañadas de una reflexión respecto al tema de la sesión y el resultado de su ejecución.

El repositorio “Modulo Programación Básica en JAVA” debe ser público. Este repositorio, al igual que su contenido serán utilizados durante todo el módulo como un “portafolio digital” a través del cual se llevará un registro de todo el proceso de aprendizaje.

La estructura del repositorio remoto deberá ser siempre la siguiente:

1. - nombre_de_usuario/modulo_programacion_basica_en_java

1.1 documentación (documento)

1.2 ejecución_1 (directorio)

1.2 ejecución_2 (directorio)

1.2.1 documento_1 (documento)

1.2.1 documento_2 (documento)

1.2.3 subdirectorio (directorio)

1.2.3.1 documento_1 (documento)

1.2.3.2 documento_2 (documento)

1.2.4 ...

1.3 ...

Cada estudiante deberá agregar al repositorio, un nuevo directorio denominado “ejecución” por cada nueva ejecución práctica que realice.

Además, deberá agregar comentarios a la documentación, las cuales dejaran constancia de cada nuevo directorio creado, de sus contenidos y sus comentarios asociados.

Al término del módulo el participante realizará una reflexión sobre los aprendizajes esperados alcanzados en el desarrollo del módulo. Además, los participantes crearán un artículo de reflexión final dando cuenta de una autocrítica sobre su propio proceso de aprendizaje.

Por otra parte, al finalizar el módulo, cada portafolio contendrá la documentación completa de los procesos individuales de trabajo de cada participante. Además de sus reflexiones personales, entre otras interesantes características.

Cada estudiante poseerá su propio sitio web y con este una URL propia (dirección web) a través de la cual los profesores(as), sus compañeros(as) e incluso terceros(as), podrán acceder con el objetivo compartir el contenido con el grupo y así potenciar el aprendizaje en comunidad y facilitar la coevaluación.

2. Estrategia didáctica

La estrategia didáctica consiste en que los participantes deberán Codificar piezas de software de baja/mediana complejidad en Java utilizando el paradigma de orientación a objetos. Este proyecto será trabajado a lo largo del módulo y deberán seleccionar una plataforma digital para registrar en línea sus proyectos.

La selección de una plataforma digital de creación de sitios web y la posterior publicación de sus trabajos clase a clases, permitirán:

- A. Que cada estudiante domine y potencie su manejo de herramientas digitales.
- B. Que el portafolio se adapte al entorno de trabajo de un proyecto de programación en JAVA
- C. Que cada estudiante maneje un proceso de documentación de su proceso de aprendizaje propio, en un entorno similar al que utilizaría en un caso profesional.
- D. Que cada estudiante reflexione sobre las clases y genere contenido original a través de reflexiones personales en su propio espacio
- E. Facilitar el acceso a los portafolios

3. Introducción al Portafolio

El portafolio busca registrar las actividades y progresos de los participantes, para ello se ha definido desarrollar un proyecto en varias etapas, el cual estará disponible en la plataforma GitHub. Dicho sitio será la bitácora del portafolio y registrará todos los avances de los participantes.

La intención de este tipo de portafolios es acercar al participante a las tecnologías digitales y la gestión de proyectos que la Industria TI.

El punto de partida del participante será: Reconocer la estructura de un algoritmo para comenzar a Codificar piezas de software de baja/mediana complejidad en Java utilizando el paradigma de orientación a objetos

Objetivos que busca alcanzar el participante a lo largo del portafolio:

1. Estructurar un algoritmo lógico utilizando estructuras de control y expresiones para dar solución a un problema de baja complejidad
2. Codificar una pieza de software utilizando el lenguaje Java de acuerdo a las buenas prácticas y convenciones para resolver un problema de baja complejidad
3. Codificar una pieza de software utilizando el paradigma de Orientación a Objetos para resolver un problema de baja complejidad
4. Codificar una pieza de software en lenguaje Java utilizando principios básicos de diseño orientado a objetos para resolver un problema de baja complejidad
5. Construir una suite de pruebas unitarias en lenguaje Java utilizando JUnit para asegurar el buen funcionamiento de una pieza de software

Contenidos y aprendizajes que será desarrollados a lo largo del portafolio:

Aprendizajes	Contenidos
1. Estructurar un algoritmo lógico utilizando estructuras de control y expresiones para dar solución a un problema de baja complejidad	<ul style="list-style-type: none">• Algoritmos:• Estructuras de control condicional.• Teoría de Conjuntos y Lógica Preposicional.• Estructuras de control repetitivas
2. Codificar una pieza de software utilizando el lenguaje Java de acuerdo a las buenas prácticas y convenciones para resolver un problema de baja complejidad	<ul style="list-style-type: none">• El Entorno Java para la programación:• Tipos de datos en Java.• Operadores en Java.• Sentencias de control.• Ciclos en Java.• Arreglos y Colecciones.

<p>3. Codificar una pieza de software utilizando el paradigma de Orientación a Objetos para resolver un problema de baja complejidad</p>	<ul style="list-style-type: none"> • Importancia de la orientación a objetos en la programación. • Clases y Objetos. • Atributos de una clase y Estado de un objeto. • Métodos de una clase y Comportamiento de un objeto. • Diagramas de Clases
<p>4. Codificar una pieza de software en lenguaje Java utilizando principios básicos de diseño orientado a objetos para resolver un problema de baja complejidad</p>	<ul style="list-style-type: none"> • Herencia de clases. • Interfaces. • Polimorfismo. • Representando polimorfismo en un diagrama de clases
<p>5. Construir una suite de pruebas unitarias en lenguaje Java utilizando JUnit para asegurar el buen funcionamiento de una pieza de software</p>	<ul style="list-style-type: none"> • Pruebas Unitarias en Java • Introducción a JUnit. • Creando un Caso de Prueba. • Integración de JUnit en Eclipse. • El Desarrollo Dirigido por Test (TDD)

4. Temas Centrales del Portafolio

La herramienta a utilizar para la creación de este portafolio será un repositorio remoto. Para esto se deberá utilizar la plataforma que se recomienda a continuación.

PLATAFORMAS SUGERIDAS	URL
GitHub	https://github.com/

Cada alumno creará su propio repositorio remoto y lo utilizará como “portafolio digital” llevando registro de todo su proceso de aprendizaje.

Actualizaciones del repositorio

Cada nueva ejecución deberá ser acompañada con la creación de un nuevo directorio en el repositorio, junto con la actualización de la documentación.

Los nuevos directorios contendrán los documentos relacionados a cada ejecución, mientras que la documentación se actualizará con notas y descripciones por cada nueva ejecución.

Antes de finalizar el módulo, cada estudiante deberá también crear una sección final con reflexiones personales sobre los contenidos cubiertos hasta ese momento.

Estructura de los directorios (1 por ejecución):

Cada nuevo directorio creado en el repositorio remoto deberá contener los siguientes elementos:

1. Título descriptivo.
2. Texto descriptivo de la o las ejecuciones realizada en esta actualización.
3. Documentos con las evidencias de la o las ejecuciones realizadas.

Estructura de la documentación (1 por ejecución):

Cada nueva actualización del repositorio remoto deberá estar acompañada de una actualización en la documentación, la cual deberá contener los siguientes elementos:

1. Título descriptivo.
2. Descripción detallada de lo que incluye la nueva actualización (Contenidos del nuevo directorio asociado)
3. Reflexión personal sobre la ejecución realizada. (Mínimo 80 palabras).

Estructura de la reflexión final (1 por módulo):

Al finalizar el curso, cada estudiante deberá crear una reflexión que debe ser publicada en el mismo documento de la documentación y que deberá contener los siguientes elementos:

1. Título.
2. Síntesis de su proceso de aprendizaje personal.
3. Momentos clave de su proceso de aprendizaje.
4. Análisis personal de sus habilidades ligadas a su proceso de aprendizaje (antes y después del curso).
5. Autocrítica.
6. Reflexión Final.

Por otro lado, al término de cada aprendizaje esperado, el participante en base a la evidencia registrada en los artículos deberá explicar el aprendizaje que creer haber conseguido. Esta reflexión será retroalimentada por el facilitador y determinarán puntos de mejora.

Aprendizajes Esperados	Aprendizajes conseguidos
1. Estructurar un algoritmo lógico utilizando estructuras de control y expresiones para dar solución a un problema de baja complejidad	
2. Codificar una pieza de software utilizando el lenguaje Java de acuerdo a las buenas prácticas y convenciones para resolver un problema de baja complejidad	
3. Codificar una pieza de software utilizando el paradigma de Orientación a Objetos para resolver un problema de baja complejidad	
4. Codificar una pieza de software en lenguaje Java utilizando principios básicos de diseño orientado a objetos para resolver un problema de baja complejidad	
5. Construir una suite de pruebas unitarias en lenguaje Java utilizando JUnit para asegurar el buen funcionamiento de una pieza de software	

5. Apartado de Cierre

Al cierre de cada módulo, los participantes tendrán que generar una síntesis del aprendizaje alcanzado en el módulo. Para lo cual, los participantes deberán reflexionar sobre:

1. En el proceso de aprendizaje logré aplicar metodologías de aprendizaje activo.
2. El portafolio de Proyecto me permitió registrar el progreso
3. El proyecto alcanzó el nivel de desarrollo esperado para este módulo

Posterior a la reflexión, el participante deberá registrar su compromiso para mejorar en el portafolio de proyecto de la siguiente unidad.

6. Definición de plataforma digital o herramienta que se utilizará para publicación y trabajo del mismo

Para llevar un registro de los diversos avances de los portafolios del curso, por cada sesión se actualizará la siguiente tabla:

Ejemplo de tabla de registro de bitácoras

ESTUDIANTE	URL REPOSITORIO	Clase 1 00/00/00	Clase 2 00/00/00	Clase 3 00/00/00	...	Total (24)
Nombre y apellido alumno	github.com/usuario/repositorio	✓	✓	✗		2 de 24
Nombre y apellido alumno	github.com/usuario/repositorio	✗	✗	✗		0 de 24
Nombre y apellido alumno	github.com/usuario/repositorio	✓	✓	✓		2 de 24
...						

La tabla de registro del curso será de libre acceso y reunirá, en cada nueva sesión, los avances de cada uno de los alumnos.

7. Instrumento evaluativo asociado al portafolio de proyecto

La evaluación del portafolio de proyecto será mediante una Escala de apreciación con una escala de 1 a 4.

Escala numérica de evaluación:

Escala	Descripción
1	No logra un trabajo que cumpla los objetivos, debó mejorar.
2	Logra un trabajo regular
3	Logra un buen trabajo
4	Logra un excelente trabajo

Indicadores	4	3	2	1
Utiliza estructuras de control condicional y expresiones lógicas				
Utiliza estructuras de control repetitivas				
Utiliza estructuras de arreglos unidimensionales y bidimensionales				
Representa un algoritmo utilizando diagramas de flujos				
Estructura un algoritmo utilizando pseudocódigo				
Utiliza tipos de datos, operadores, sentencias de control, ciclos y colecciones de Java				
Utiliza clases predefinidas y bibliotecas del entorno Java				
Analiza el código de programación Java utilizando las herramientas de depuración disponibles en el IDE				
Realiza la documentación del código fuente				
Realiza codificación mantenible				
Construir una aplicación que se ejecuta en la consola de comandos utilizando el lenguaje Java				
Codifica una clase en lenguaje Java utilizando constructores, accesadores y mutadores				
Codifica clases en lenguaje Java utilizando colaboración y composición				
Codifica clases utilizando el lenguaje Java a partir de un Diagrama de Clases				
Representa en un Diagrama de Clases el modelo de clases				
Codifica clases en lenguaje Java utilizando herencia				
Codifica clases en lenguaje Java utilizando interfaces				
Codifica clases en lenguaje Java utilizando principios de diseño para obtener componentes con alta cohesión y bajo acoplamiento				
Construye Casos de Prueba utilizando JUnit				
Construye una Suite de Pruebas utilizando JUnit				
Ejecuta una suite de pruebas utilizando JUnit				

Construye una pieza de software utilizando el método de desarrollo dirigido por pruebas (TDD)				
Aplica de forma correcta los conceptos revisados				
Aplica las pautas establecidas por el facilitador				
Registra evidencias de los avances de su proyecto en la bitácora definida.				
Acepta críticas sobre puntos de mejora del proyecto				
Acepta la cooperación de compañeros para mejorar su proyecto				