

## Problema 2. Calculando la nota definitiva

El profesor de Matemáticas necesita un sistema sencillo que calcule la nota final de cada estudiante a partir de cuatro componentes evaluativos: Proyecto final (30 %), Exposición (20 %), Talleres (30 %) y Parcial (20 %). Para cada estudiante se registran las calificaciones de estos componentes en la misma escala del curso (p. ej., 0–100 o 0–5). El sistema debe validar que los valores ingresados estén dentro de los límites permitidos por la escala y debe producir la nota final ponderada como suma de cada componente multiplicado por su porcentaje. La salida debe presentarse en la misma escala y con una presentación legible (por ejemplo, con dos decimales).

El objetivo es modelar el problema con orientación a objetos. El programa debe encapsular los datos del estudiante y los componentes de evaluación, exponer operaciones para registrar/consultar las calificaciones y calcular la nota final aplicando los pesos fijos (30/20/30/20). No se consideran recuperaciones ni componentes adicionales. El diseño debe quedar preparado para, en el futuro, cambiar la escala (0–100, 0–5, etc.) o ajustar porcentajes, sin modificar la lógica central de cálculo. Un ejemplo de ejecución se muestra a continuación:

Ingrese la nota del proyecto: 4,5  
Ingrese la nota de la exposicio n: 3,6  
Ingrese la nota de talleres: 5  
Ingrese la nota del parcial: 3



La nota final del estudiante es 4,17

**Tabla de los R#**

<b>R#</b>	<b>Descripción</b>
<b>R1</b>	El sistema debe permitir registrar el nombre del estudiante
<b>R2</b>	El sistema debe permitir registrar la nota del proyecto final, validando que esté dentro de la escala
<b>R3</b>	El sistema debe permitir registrar la nota de la exposición, validando que esté dentro de la escala
<b>R4</b>	El sistema debe permitir registrar la nota de los talleres, validando que esté dentro de la escala
<b>R5</b>	El sistema debe permitir registrar la nota de los parciales, validando que esté dentro de la escala
<b>R6</b>	El sistema debe calcular la nota final ponderada aplicando: $(0.3 \times \text{Proyecto}) + (0.2 \times \text{Exposición}) + (0.3 \times \text{Talleres}) + (0.2 \times \text{parcial})$
<b>R7</b>	El sistema debe mostrar un reporte de la nota final en la misma escala con 2 decimales

## Vocabulario del dominio y clases candidatas

<b>Término normalizado</b>	<b>Tipo</b>	<b>Definición (1 línea, lenguaje del dominio)</b>	<b>Sinónimos consolidados</b>	<b>¿Clase/Atributo/Operación candidata?</b>
<b>Estudiante</b>	Entidad	Persona que recibe las calificaciones	Aprendiz	Nombre_ Notas_
<b>Notas</b>	Atributo	Calificación numérica del estudiante por cada componente	Calificación	Atributo <n0,n1,n2,n3>
<b>Componente Evaluativo</b>	Entidad	Estructura académica sobre la cual se califica el curso	Rúbrica	Método (CalcularNotaFinal() )
<b>Proyecto final</b>	valor/porcentual	Valor porcentual (30%)	Plan final	operación
<b>Exposición</b>	valor/porcentual	Valor porcentual (20%)	Presentación	operación
<b>Talleres</b>	valor/porcentual	Valor porcentual (30%)	Trabajos	operación
<b>parciales</b>	valor/porcentual	Valor porcentual (20%)	Exámenes	operación
<b>Escala</b>	Constante	Límites permitidos de las notas	Rango	operación
<b>Nota final</b>	Operación	Suma de los valores obtenidos por cada nota	Resultado	Atributo (nota_final)
<b>Reporte de Resultados</b>	Entidad	Información resultante del monitoreo de notas de estudiantes	Informe	Método (ReporteResultados())
<b>Validación de notas</b>	Operación	Comprobación de que estén en el rango válido	Control	

## Tabla CRC

### Estudiante

Tabla CRC	Contenido a completar
<b>Clase:</b> Estudiante	
<b>Responsabilidades (qué hace):</b>	
Almacenar las calificaciones	1) Almacenamiento de datos (Nombre/Notas)
<b>Colaboradores (a quién le pide):</b>	
Evaluación, Escala	2) Envía las notas a la evaluación
<b>Mensajes clave:</b>	
evaluacion.calcularNotas(notas)	
escala.validarNota()	

## Tabla CRC

### Evaluación

Campo	Contenido a completar
<b>Clase:</b> ComponenteEvaluativo	
<b>Responsabilidades (qué hace):</b>	
Calcula la nota final del estudiante en base a las ponderaciones registradas.	1) Recibe las notas registradas del estudiante
<b>Colaboradores (a quién le pide):</b>	
Estudiante	3) Almacena la calificación y el peso de la misma del estudiante
<b>Mensajes clave:</b>	
evaluacion.calcularNota()	
evaluacion.reporteEstudiante()	

## Tabla CRC

### Escala

Campo	Contenido a completar
<b>Clase:</b> Reporte	
<b>Responsabilidades (qué hace):</b>	
Validar notas	1) Valida la nota recibida del estudiante
<b>Colaboradores (a quién le pide):</b>	
Estudiante	
<b>Mensajes clave:</b>	
escala.validarNota()	

## Tabla CRC

### Componente

Campo	Contenido a completar
<b>Clase:</b> Componente	
<b>Responsabilidades (qué hace):</b>	
Almacena las notas y pesos del componente, calcula la nota ponderada del componente	1)
<b>Colaboradores (a quién le pide):</b>	
Estudiante, Evaluación	
<b>Mensajes clave:</b>	
componente.calcularNotasPonderadas	

# EPS

## Estudiante

<b>Estudiante</b>	
<b>Campo</b>	<b>Contenido a completar</b>
<b>Clase:</b> Estudiante	
<b>Métodos:</b>	
<b>Propósito (una línea)</b>	Almacenar datos de las notas y nota final
<b>E/P/S</b>	
<b>Entradas (datos requeridos)</b>	—
<b>Proceso (pasos, decisiones, fórmulas si aplica)</b>	1)
	2)
	3)
<b>Salidas (resultado observable)</b>	—
<b>Contrato PRE/POST</b>	
<b>PRE</b>	
<b>POST</b>	



*función* calcularNota(estudiante):

    suma  $\leftarrow$  0

    para i desde 0 hasta tamaño(estudiante.notas) - 1:

        suma  $\leftarrow$  suma + (estudiante.notas[i] \* pondComponentes[i])

    estudiante.notaFinal  $\leftarrow$  suma

    retornar estudiante.nota

Fin

*pseudocódigo*

*funcion* validarNota(nota):

    si nota < mínimo o nota > máximo:

        retornar FALSO

    retornar VERDADERO

# EPS

## Escala

Escala	
Campo	Contenido a completar
<b>Clase:</b> escala	
<b>Método:</b>	
<b>Propósito (una línea)</b>	
<b>E/P/S</b>	
<b>Entradas (datos requeridos)</b>	- Notas <proyecto final() talleres() exposición() parciales()>
<b>Proceso (pasos, decisiones, fórmulas si aplica)</b>	1) Notas: 0-100 o 0-5
<b>Salidas (resultado observable)</b>	- Valores válidos entre dichos límites
<b>Contrato PRE/POST</b>	
<b>PRE</b>	
<b>POST</b>	

# EPS

## Evaluación

Evaluación	
Campo	Contenido a completar
<b>Clase:</b> Evaluación	
<b>Método:</b>	
<b>Propósito (una línea)</b>	
<b>E/P/S</b>	
<b>Entradas (datos requeridos)</b>	– escala()
<b>Proceso (pasos, decisiones, fórmulas si aplica)</b>	1) Se debe presentar en la misma escala y con una presentación legible
	2) $\text{Nota final} \leftarrow \text{proyecto final} * 0.3 + \text{exposicion} * 0.2 + \text{taller es} * 0.3 + \text{parciales} * 0.2$
	3)
<b>Salidas (resultado observable)</b>	
<b>Contrato PRE/POST</b>	
<b>PRE</b>	
<b>POST</b>	

# EPS

## Componente

Evaluación	
Campo	Contenido a completar
Clase: Componente	
Método:	calcularNotaPonderada
Propósito (una línea)	Almacenar los pesos y notas, Calcular nota ponderada
E/P/S	
Entradas (datos requeridos)	– notas, pesos,
Proceso (pasos, decisiones, fórmulas si aplica)	1) multiplicar las notas por los pesos
	2) devolver nota ponderada
	3)
Salidas (resultado observable)	
Contrato PRE/POST	
PRE	Estudiante con notas creado
POST	Cálculo de la nota final

## Diagrama de clase

