Taller Integrador: Patrones de diseño

# Sistema de Requerimientos Académicos.

# Objetivos

* Evaluar que patrones de diseño se deben utilizar en el desarrollo de un sistema.
* Aplicar varios patrones de diseño dentro de un mismo sistema.

# Requerimientos del sistema:

Se desea desarrollar un sistema para el Ministerio de Educación, mediante el cual estudiantes de distintas Instituciones Educativas (escuelas, colegios y universidades) puedan consultar información relacionada a sus calificaciones.

El sistema debe utilizar patrones de diseño y buenas prácticas de programación. Además, debe ejecutarse en una estación (PC de escritorio) disponible en la oficina principal de dicho ministerio.

Para mostrar las opciones del sistema primero debe elegir la Institución Educativa donde está matriculado el estudiante y mostrar los datos almacenados de esa institución. Pero, como programa piloto se desea soportar primero las siguientes Unidades Educativas del Milenio, pero después se deberán agregar otras Instituciones de Educativas:

Luego debe mostrar un menú con las opciones permitidas por la institución Educativa.

Unidad Educativa del Milenio JATUN KURAKA:

* 1. Imprimir promedio de calificaciones de estudiante con escala extranjera.
  2. Elegir otra institución.

Unidad Educativa del Milenio PROF. CONSUELO BENAVIDES:

1. Imprimir el certificado de estudiante matriculado.
2. Elegir otra institución.

Unidad Educativa del Milenio SUMAK YACHANA WASI:

1. Imprimir el certificado de estudiante matriculado.
2. Imprimir promedio de calificaciones de estudiante con escala extranjera.
3. Elegir otra institución.

Las opciones mostradas dependen de ciertos convenios firmados entre las instituciones educativas y el ministerio de educación. Por lo tanto, estas funcionalidades son opcionales para las instituciones educativas (Use datos estáticos de prueba).

Para imprimir el certificado de estudiante matriculado, se debe pedir el número de matrícula del estudiante y verificarlo con el listado de estudiantes matriculados en la institución elegida. El certificado debe mostrar la información de la institución junto con el nombre completo, la matrícula y el promedio del estudiante.

Por otro lado, actualmente se tiene un listado de países que han firmado convenios con el ministerio de educación para convalidar las calificaciones de los estudiantes, siempre y cuando se utilicen las escalas provistas por dichos países. Pero, en el mediano o corto plazo se espera firmar más convenios con otros países para poder realizar la convalidación de calificaciones de estudiantes que desean ir a estudiar en el extranjero.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nro | País | Escala del país | Aprobado | Muy Bueno | Sobresaliente | Excelente |
| 1 | Ecuador | 2.0 🡪 10.0 | 7.0 🡪 7.9 | 8.0 🡪 8.9 | 9.0 🡪 9.5 | 9.5 🡪 10.0 |
| 2 | Alemania | 4.0 🡪 1.0 | 4.0 🡪 2.1 | 2.0 🡪 1.4 | 1.3 🡪 1.1 | 1.0 |
| 3 | Estados Unidos de América | D+ 🡪 A+ | D+ 🡪 C+ | B- 🡪 B+ | A- 🡪 A+ | A |

# Desarrollar

1. Indique que patrones podrían servir dentro del desarrollo de este sistema. (explique)
   1. **Creacionales.**

Como lo que se requiere es la construcción de varios documentos con información específica de ciertos estudiantes, nos serviría lo que es el patrón de diseño **Builder**, puesto a que este separa la construcción de un objeto complejo de su representación final, de tal manera que el mismo proceso de construcción pueda crear representaciones diferentes, como lo son los certificados, actas de calificaciones, promedios con escalas extranjeras, etc.

* 1. **Estructurales.**

Aquí cabe recalcar que no se sabe si nuestro programa permanecerá fijo en una sola maquina con un solo sistema operativo o estará en traslado constante, por lo que sería ideal crearle o agregarle un patrón adicional llamado **Adapter** para que si esto ocurre, nuestro programa pueda adaptarse sin problemas a la nueva máquina, este patrón utiliza una clase existente cuya interfaz no se corresponde con lo que necesita para su funcionamiento, o simplemente crea una clase reutilizable que coopera con las clases imprevistas, ósea que no necesariamente tienen interfaces compatibles.

* 1. **De Comportamiento.**

En una parte de este documento se dice que “en el mediano o corto plazo se espera firmar más convenios con otros países para poder realizar la convalidación de calificaciones de estudiantes que desean ir a estudiar en el extranjero”, por lo que habrá que ver la evolución de los estudiantes, o hacerle seguimientos de estados, si lo que se quiere es saber de manera automática que estudiantes serán aptos para estudiar en el extranjero lo que se recomendaría es aplicar el patrón de comportamiento **Observer** cuya función principal es notificar los cambios de estados a objetos dependientes, en este caso las notas y/o promedios de los estudiantes, y con esta respuesta enviar notificaciones de estados a los objetos correspondientes.

1. Analice el diagrama de clases adjunto e indique ventajas y desventajas del uso de cada patrón de diseño utilizado.

Ventajas

* Como se está utilizando el patrón Builder si se requiere generar otro tipo de dato solo se implementaría esa funcionalidad y hacer que el Builder construya esta nueva funcionalidad.
* Gracias al patrón Decorator si se desea agregar una nueva funcionalidad se puede implementarla sin que esta afecte a las demás funcionalidades anteriores.
* Como se utiliza el patrón Strategy para controlar las escalas de puntajes que los estudiantes pueden tener si se desea agregar un nuevo país solo se generaría la escala de dicho país y se la asociaría al patrón sin ningún inconveniente.

Desventajas

* Debido a que se está utilizando el Decorator se pueden implementar varias funcionalidades, pero esto causaría que el sistema sea un poco complejo de aprender y muy difícil de depurar.
* Debido a la estructura que maneja el patrón Strategy se pueden aumentar varios objetos pero esto causaría que el sistema tenga una menor eficiencia, además pueden haber complicaciones en la comunicación entre el contexto y las estrategias aplicadas al sistema.

1. Proponga un nuevo diagrama de clases que ayude a manejar las fallas del diagrama anterior.
2. Implemente su solución.