**Nombre:** Fernando Eliceo Huilca Villagómez **Fecha:** 04/06/2025 **Curso:** GR2SW

**Módulo: Ejercicios y Evaluación**

**Casos de usos involucrados:**

**CASO DE USO: Realizar Ejercicios Interactivos**

**Actor principal:** Estudiante **Objetivo en contexto:** Resolver ejercicios prácticos con corrección automática. **Precondiciones:** Debe estar dentro de una lección activa. **Disparador:** El estudiante inicia un ejercicio desde la lección.

**Escenario:**

1. Se carga un ejercicio con instrucciones.

2. El estudiante escribe o selecciona la respuesta.

3. El sistema evalúa la respuesta.

4. Se muestra retroalimentación inmediata.

**Excepciones:**

1. Si el código tiene errores, se ofrece diagnóstico.

2. Si hay fallo de red, se guarda en cola y se evalúa más tarde.

**Prioridad:** MUST **Cuándo estará disponible:** Primer incremento. **Frecuencia de uso:** Frecuente. **Canal para el actor:** Editor web / móvil. **Actores secundarios:** Motor de evaluación. **Canales para los actores secundarios:** Motor de retroalimentación.

**Aspectos pendientes:**

1. ¿Puede el usuario pausar o guardar la resolución del ejercicio?

2. ¿Se mostrarán pistas o soluciones si el usuario falla repetidamente?

3. ¿Se deben permitir múltiples intentos por ejercicio?

**Desarrollo de la práctica:**

El modelado basado en clases representa los objetos que manipulará el sistema, las operaciones (también llamadas *métodos* o *servicios*) que se aplicarán a los objetos para efectuar la manipulación, las relaciones (algunas de ellas jerárquicas) entre los objetos y las colaboraciones que tienen lugar entre las clases definidas. Pag 142

**Análisis gramatical:**

problema de una aplicación de software, las clases (y objetos) son má difíciles de concebir. Se comienza por identificar las clases mediante el análisis de los escenarios de uso desarrollados como parte del modelo de requerimientos y la ejecución de un “análisis gramatical” [Abb83] sobre los casos de uso desarrollados para el sistema que se va a construir. Las clases se determinan subrayando cada sustantivo o frase que las incluya para introducirlo en una tabla simple. Deben anotarse los sinónimos. Si la clase (sustantivo) se requiere para implementar una solución, entonces forma parte del espacio de solución; de otro modo, si sólo es necesaria para describir la solución, es parte del espacio del problema. Pag 142

Un sustantivo es una palabra que nombra o designa personas, animales, lugares, cosas, ideas, sentimientos, o cualquier entidad, tanto real como ficticia. (web)

Caso de Uso (narrativa):   
El **sistema BitByBit** permite que un **estudiante** resuelva **ejercicios** **interactivos** dentro de una **lección** activa.

Cuando comienza un **ejercicio**, se muestra una **instrucción** clara que guía lo que debe hacerse.

El **estudiante** responde escribiendo código o seleccionando una opción, según el **tipo de ejercicio**.

Una vez enviada la **respuesta**, el **sistema** la analiza y determina si es correcta o no.

Si la respuesta es incorrecta, el **sistema** proporciona **retroalimentación** inmediata para ayudar al **estudiante** a mejorar.

En caso de que el **código** enviado tenga **errores**, el sistema ofrece un diagnóstico para facilitar su corrección.

Si ocurre un **problema** de conexión a la **red**, la respuesta se guarda y será evaluada más tarde sin que el estudiante pierda su progreso.

El **estudiante** puede continuar con el siguiente **ejercicio**, intentar de nuevo, o revisar la **retroalimentación** antes de avanzar.

|  |  |  |
| --- | --- | --- |
| **Clase potencial** | **Clasificación** **general** | **Descripción** |
| Estudiante | Rol - Entidad Externa | Rol que desempeña una persona que interactúa con el sistema |
| Ejercicio Interactivo | Cosa | Dominio de información - Estructura principal que maneja la lógica de evaluación |
| Respuesta | Cosa | Dominio de información - Información generada por el estudiante |
| Lección Activa | Cosa | Dominio de información - Unidad organizacional que contiene ejercicios |
| Solución | Cosa | Dominio de información |
| Resultado | Ocurrencia o evento | Ocurrencia de un evento de evaluación |
| Pista | Ocurrencia o evento | Sucede dentro del contexto |
| Instrucción | Ocurrencia o evento | Sucede dentro del contexto |
| Retroalimentación | Cosa | Dominio de información |
| Diagnostico | Cosa | Dominio de información - Información especializada derivada de errores |
| Motor Evaluación | Estructura | Definen clases relacionada - Clase auxiliar para evaluación |
| Motor retroalimentación | Estructura | Definen clases relacionada - Clase auxiliar para generar feedback |
| Errores | Cosas | Dominio de información |
| Red | Entidades externas | Produce o  consume la información |
| Cola | Estructura | Definen clases relacionada |

Como afirma Cashman [Cas89]:

“el intento de la orientación a objetos es contener, pero mantener separados, los datos y las

operaciones sobre ellos”.

Coad y Yourdon [Coa91] sugieren seis características de selección que deben usarse cuando

se considere cada clase potencial para incluirla en el modelo de análisis:

**1.** *Información retenida.* La clase potencial será útil durante el análisis sólo si debe recordarse

la información sobre ella para que el sistema pueda funcionar.

**2.** *Servicios necesarios*. La clase potencial debe tener un conjunto de operaciones identificables

que cambien en cierta manera el valor de sus atributos.

**3.** *Atributos múltiples.* Durante el análisis de los requerimientos, la atención debe estar en

la información “principal”; en realidad, una clase con un solo atributo puede ser útil durante

el diseño, pero es probable que durante la actividad de análisis se represente mejor

como un atributo de otra clase.

**4.** *Atributos comunes*. Para la clase potencial se define un conjunto de atributos y se aplican

éstos a todas las instancias de la clase.

**5.** *Operaciones comunes*. Se define un conjunto de operaciones para la clase potencial y

éstas se aplican a todas las instancias de la clase.

**6.** *Requerimientos esenciales.* Las entidades externas que aparezcan en el espacio del problema

y que produzcan o consuman información esencial para la operación de cualquier

solución para el sistema casi siempre se definirán como clases en el modelo de

requerimientos.

**Para que se considere una clase legítima para su inclusión en el modelo de requerimientos,**

**un objeto potencial debe satisfacer todas (o casi todas) las características anteriores. La decisión**

**de incluir clases potenciales en el modelo de análisis es algo subjetiva, y una evaluación posterior**

**tal vez haga que un objeto se descarte o se incluya de nuevo.**

1. **Estudiante (Rol/Entidad Externa)**
   * **✅ Información retenida (datos del usuario)**
   * **✅ Servicios necesarios (iniciar ejercicios, enviar respuestas)**
   * **✅ Atributos múltiples (ID, nombre, progreso, etc.)**
   * **✅ Atributos comunes (a todos los estudiantes)**
   * **✅ Operaciones comunes (métodos compartidos)**
   * **✅ Requerimientos esenciales (actor principal)  
     Conclusión: CLASE VÁLIDA**
2. **Ejercicio Interactivo (Cosa)**
   * **✅ Información retenida (estado, preguntas, solución)**
   * **✅ Servicios necesarios (evaluar respuestas, mostrar retroalimentación)**
   * **✅ Atributos múltiples (ID, tipo, dificultad, etc.)**
   * **✅ Atributos comunes (a todos los ejercicios)**
   * **✅ Operaciones comunes (métodos compartidos)**
   * **✅ Requerimientos esenciales (núcleo del sistema)  
     Conclusión: CLASE VÁLIDA**
3. **Respuesta (Cosa)**
   * **✅ Información retenida (contenido, corrección)**
   * **❌ Servicios necesarios (solo almacena datos)**
   * **✅ Atributos múltiples (ID, contenido, fecha)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes (pocas operaciones propias)**
   * **✅ Requerimientos esenciales  
     Conclusión: CLASE VÁLIDA (pero podría ser estructura interna de Ejercicio)**
4. **Lección Activa (Cosa)**
   * **✅ Información retenida (estado, progreso)**
   * **✅ Servicios necesarios (activar/desactivar, gestionar ejercicios)**
   * **✅ Atributos múltiples (ID, título, duración)**
   * **✅ Atributos comunes**
   * **✅ Operaciones comunes**
   * **✅ Requerimientos esenciales  
     Conclusión: CLASE VÁLIDA**
5. **Solución (Cosa)**
   * **✅ Información retenida (respuesta correcta)**
   * **❌ Servicios necesarios (solo almacena datos)**
   * **❌ Atributos múltiples (normalmente un solo campo)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes**
   * **✅ Requerimientos esenciales  
     Conclusión: NO ES CLASE (mejor como atributo de Ejercicio)**
6. **Resultado (Evento)**
   * **✅ Información retenida (éxito/fracaso, puntuación)**
   * **❌ Servicios necesarios (registra datos, no cambia)**
   * **✅ Atributos múltiples (correcto, puntuación, tiempo)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes**
   * **✅ Requerimientos esenciales  
     Conclusión: CLASE VÁLIDA (pero podría ser estructura de Ejercicio)**
7. **Pista (Evento)**
   * **✅ Información retenida (contenido de ayuda)**
   * **❌ Servicios necesarios**
   * **❌ Atributos múltiples (normalmente texto simple)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes**
   * **❌ Requerimientos esenciales (complementario)  
     Conclusión: NO ES CLASE (atributo de Ejercicio o Retroalimentación)**
8. **Instrucción (Evento)**
   * **✅ Información retenida (texto explicativo)**
   * **❌ Servicios necesarios**
   * **❌ Atributos múltiples (texto + posible imagen)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes**
   * **✅ Requerimientos esenciales  
     Conclusión: NO ES CLASE (atributo de Ejercicio)**
9. **Retroalimentación (Cosa)**
   * **✅ Información retenida (mensajes al usuario)**
   * **❌ Servicios necesarios**
   * **✅ Atributos múltiples (tipo, contenido, nivel)**
   * **✅ Atributos comunes**
   * **❌ Operaciones comunes**
   * **✅ Requerimientos esenciales  
     Conclusión: CLASE VÁLIDA (pero podría fusionarse con Resultado)**
10. **Diagnóstico (Cosa)**
    * **✅ Información retenida (análisis de errores)**
    * **✅ Servicios necesarios (generar sugerencias)**
    * **✅ Atributos múltiples (tipo error, gravedad, solución)**
    * **✅ Atributos comunes**
    * **✅ Operaciones comunes**
    * **✅ Requerimientos esenciales  
      Conclusión: CLASE VÁLIDA**
11. **Motor Evaluación (Estructura)**
    * **✅ Información retenida (configuración, reglas)**
    * **✅ Servicios necesarios (evaluar código, comparar respuestas)**
    * **✅ Atributos múltiples (versión, lenguajes soportados)**
    * **✅ Atributos comunes**
    * **✅ Operaciones comunes**
    * **✅ Requerimientos esenciales  
      Conclusión: CLASE VÁLIDA**
12. **Motor Retroalimentación (Estructura)**
    * **✅ Información retenida (plantillas, estrategias)**
    * **✅ Servicios necesarios (generar mensajes personalizados)**
    * **✅ Atributos múltiples (estilos, niveles)**
    * **✅ Atributos comunes**
    * **✅ Operaciones comunes**
    * **✅ Requerimientos esenciales  
      Conclusión: CLASE VÁLIDA**
13. **Errores (Cosa)**
    * **✅ Información retenida (tipos de errores)**
    * **❌ Servicios necesarios (datos estáticos)**
    * **❌ Atributos múltiples (normalmente enum simple)**
    * **✅ Atributos comunes**
    * **❌ Operaciones comunes**
    * **✅ Requerimientos esenciales  
      Conclusión: NO ES CLASE (mejor como enum o atributo de Diagnóstico)**
14. **Red (Entidad Externa)**
    * **❌ Información retenida (no gestiona datos)**
    * **❌ Servicios necesarios**
    * **❌ Atributos múltiples**
    * **❌ Atributos comunes**
    * **❌ Operaciones comunes**
    * **❌ Requerimientos esenciales (infraestructura, no dominio)  
      Conclusión: NO ES CLASE**
15. **Cola (Estructura)**
    * **✅ Información retenida (respuestas pendientes)**
    * **✅ Servicios necesarios (encolar, desencolar, reintentar)**
    * **✅ Atributos múltiples (elementos, estado, reintentos)**
    * **✅ Atributos comunes**
    * **✅ Operaciones comunes**
    * **✅ Requerimientos esenciales (para manejo offline)  
      Conclusión: CLASE VÁLIDA**

|  |  |  |
| --- | --- | --- |
| **Clase potencial** | **Numero de características que aplica** | **Conclusión** |
| Estudiante | 6 | Clase válida |
| Ejercicio Interactivo | 6 | Dominio de información - Estructura principal que maneja la lógica de evaluación |
| Respuesta | 4 | Dominio de información - Información generada por el estudiante |
| Lección Activa | Cosa | Dominio de información - Unidad organizacional que contiene ejercicios |
| Solución | Cosa | Dominio de información |
| Resultado | Ocurrencia o evento | Ocurrencia de un evento de evaluación |
| Pista | Ocurrencia o evento | Sucede dentro del contexto |
| Instrucción | Ocurrencia o evento | Sucede dentro del contexto |
| Retroalimentación | Cosa | Dominio de información |
| Diagnostico | Cosa | Dominio de información - Información especializada derivada de errores |
| Motor Evaluación | Estructura | Definen clases relacionada - Clase auxiliar para evaluación |
| Motor retroalimentación | Estructura | Definen clases relacionada - Clase auxiliar para generar feedback |
| Errores | Cosas | Dominio de información |
| Red | Entidades externas | Produce o  consume la información |
| Cola | Estructura | Definen clases relacionada |

**Definición de atributos.**

Para desarrollar un conjunto de atributos significativos para una clase de análisis, debe estudiarse

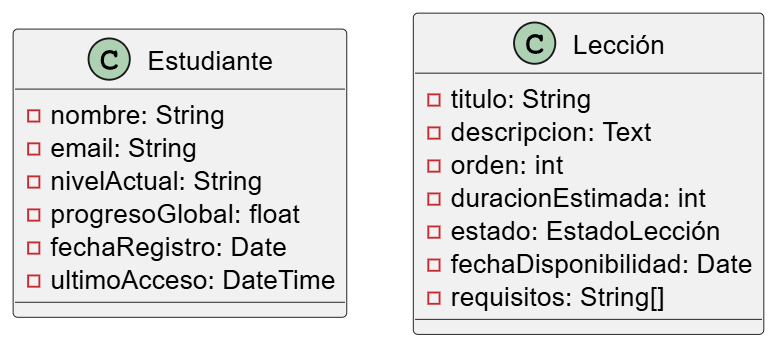
cada caso de uso y seleccionar aquellas “cosas” que “pertenezcan” razonablemente a la

clase. Además, debe responderse la pregunta siguiente para cada clase: “¿qué aspectos de los

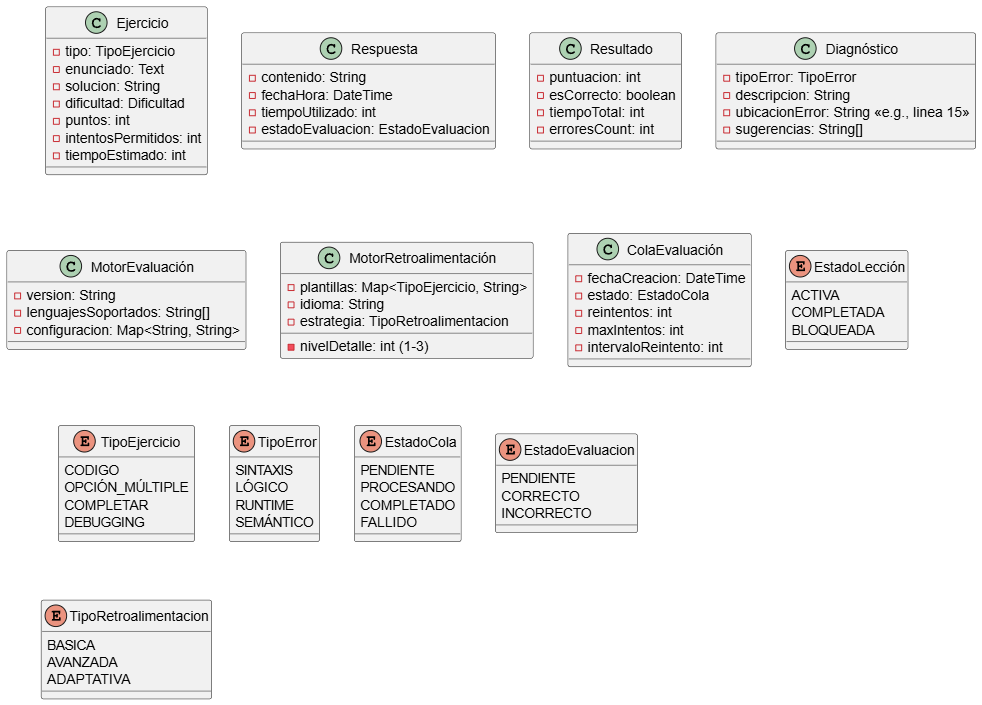
datos (compuestos o elementales) definen por completo esta clase en el contexto del problema

en cuestión?”

Otros módulos(lo hago para ocuparlo como previsualización):



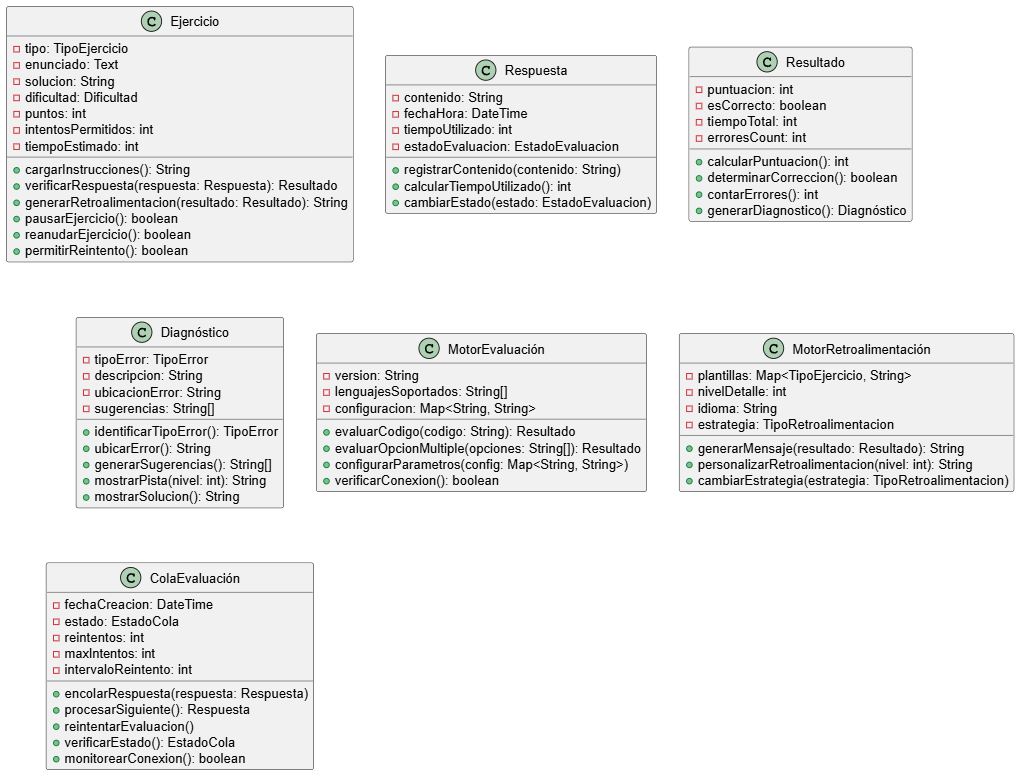
Modulo nuestro:



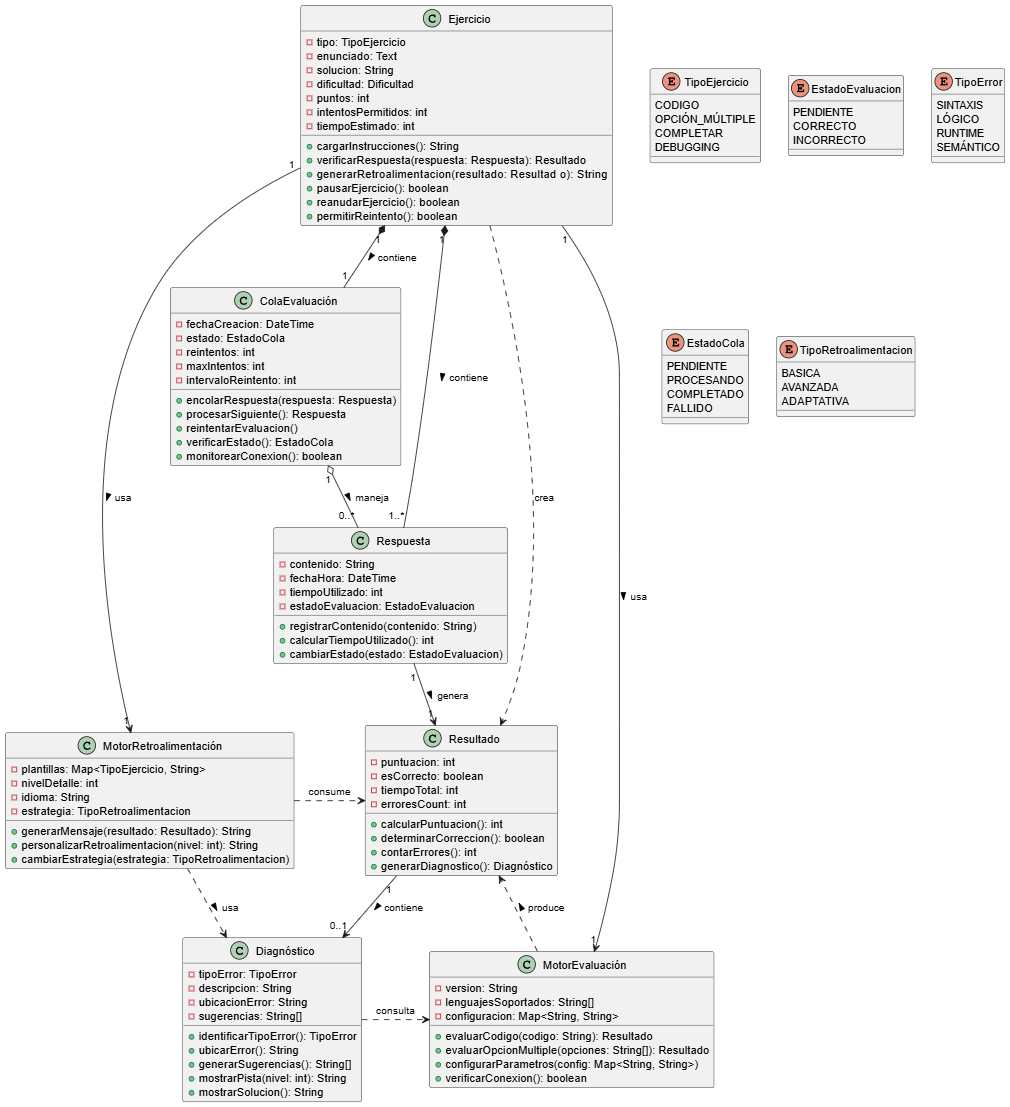
DEFINICIÓN DE MÉTODOS:

**Verbos clave en el caso de uso:**

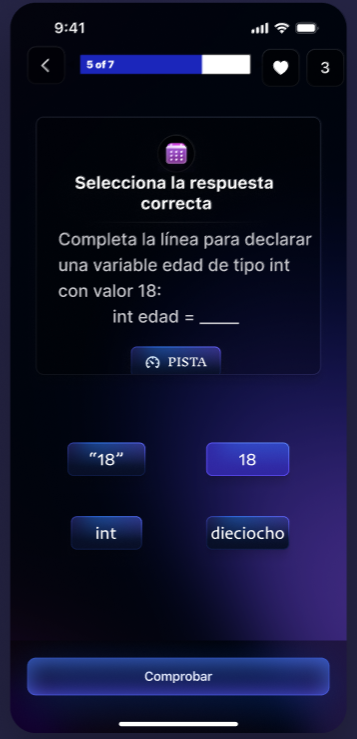
* Cargar ejercicio
* Escribir/seleccionar respuesta
* Evaluar respuesta
* Mostrar retroalimentación
* Ofrecer diagnóstico
* Guardar en cola
* Evaluar más tarde
* Pausar/guardar resolución
* Mostrar pistas/soluciones
* Permitir múltiples intentos



Relaciones entre clases:



Prototipo:



Palabras diccionario:

Respuesta: Solución propuesta por el estudiante durante el ejercicio.

Solución: Implementación correcta esperada, definida por el sistema.

Resultado: Evaluación objetiva del desempeño en el ejercicio

Retroalimentación: Información formativa para guiar el aprendizaje

Diagnostico: Identificación técnica de errores en la respuesta