

# Evaluar Examen del Curso (1, 2, 3) - EXAMEN ATC - Agregar contacto - [guardar\_examen\_old]

## ? Descripción

*Este servicio registra las respuestas de un examen enviado por un estudiante.*

La función:

1. **Recibe** el DNI del estudiante, el identificador del examen y las respuestas.
2. **Valida** que el estudiante exista en el ERP.
3. **Obtiene** la definición completa del examen (Quiz) desde el ERP.
4. **Obtiene** la información detallada de cada pregunta usando los `question_link`.
5. **Devuelve** el listado de preguntas del examen, incluyendo su estructura completa.

### ⚠ **Importante:**

Este servicio *no guarda las respuestas* del examen.

Solo valida al estudiante y retorna las preguntas del examen consultado.

---

## ? Endpoint

**POST** /guardar-examen-old

---

## ? Request Body (JSON)

```
{
  "dni": "string",
  "examen": "string",
  "respuestas": "json_string"
}
```

Ejemplo:

```
{
  "dni": "12345678",
  "examen": "QUIZ-0001",
  "respuestas": "{\"P1\":\"A\",\"P2\":\"C\"}"
}
```

## ? Seguridad

- Requiere autenticación ERP (a través de `ServiceErp()` ).
- No utiliza tokens externos.

## ? Flujo del Servicio (Explicación detallada)

### 1. Recibe datos del request:

- `$dni`
- `$examen`
- `$respuestas` (JSON string → array)

### 2. Decodifica las respuestas del examen:

```
$respuestas = json_decode($respuestas, true);
```

### 3. Busca al estudiante por DNI (solo estudiantes habilitados):

GET Student

```
filters=[["Student","enabled","=",1],["Student","dni","like","%<dni>%"]]
```

- Si no encuentra coincidencias → `[]` devuelve error:

```
{ "valor": false, "msn": "Su usuario no tiene cursos asignados" }
```

### 4. Obtiene la definición del examen (Quiz) desde el ERP:

GET Quiz/<examen>

- Trae todas las preguntas vinculadas.

## 5. Por cada pregunta del examen:

- Consulta la API:  
GET Question/{question\_link}
- Agrega la información completa al arreglo final.

## 6. Devuelve todas las preguntas consultadas.

---

# ? Response 200 – Ejemplo

```
[
  {
    "name": "QUESTION-001",
    "question": "¿Cuál es la capital de Perú?",
    "options": [
      {"option": "Lima", "is_correct": 1},
      {"option": "Cusco", "is_correct": 0}
    ]
  },
  {
    "name": "QUESTION-002",
    "question": "Seleccione la opción correcta...",
    "options": [ ... ]
  }
]
```

# ? Posibles Errores

## 1. Estudiante no existe o no está habilitado

```
{
  "valor": false,
  "msn": "Su usuario no tiene cursos asignados"
}
```

## 2. Error consultando Quiz o Preguntas

(NOT IMPLEMENTED, pero puede ocurrir internamente)

```
{
  "valor": false,
  "msn": "Error al obtener examen",
  "data": {}
}
```

### 3. Respuestas mal formateadas

Si el JSON no es válido:

```
{
  "valor": false,
  "msn": "Formato inválido de respuestas"
}
```

(Nota: El código actual **no valida** esto, pero debería.)

---

## ? Schemas usados

### Student (GET)

```
{
  "name": "string",
  "dni": "string",
  "enabled": 1
}
```

### Quiz (GET)

```
{
  "name": "string",
  "question": [
    { "question_link": "QUESTION-001" }
  ]
}
```

### Question (GET)

```
{
  "name": "string",
  "question": "string",
  "options": [
    { "option": "string", "is_correct": 0|1 }
  ]
}
```

---

# ? Lógica en pseudocódigo

```
dni = request.dni
examen = request.examen
respuestas = json_decode(request.respuestas)

student = GET Student WHERE enabled=1 AND dni LIKE %dni%

if student not found:
  return error

quiz = GET Quiz/{examen}

preguntas = []
foreach link in quiz.question:
  p = GET Question/{question_link}
  preguntas.append(p)

return preguntas
```

---

Revisión #3

Creado 2025-11-25 17:55:38 -05 por Elian

Actualizado 2025-11-26 11:08:10 -05 por Elian