



# INSTRUCTIVO DE CONEXIÓN: INTEGRACIÓN ENTRE LMS Y SIC (API GESTOR INTERMEDIO)

## Tabla de Contenido

---

Últimos Cambios Realizados al Instructivo .....	3
Introducción .....	4
Identificación de Datos a Enviar .....	5
Estructura Árbol Datos .....	8
Estructurar la Información .....	9
Recomendaciones de IDE (Entorno de Desarrollo Integrado) y Extensiones .....	9
Obtener Autenticación de "Inicio Sesión SENCE" .....	10
Prueba de Conexión .....	11
Envío de Datos a API REST .....	12
Preguntas Frecuentes .....	16
Configuración de Moodle .....	18
Identificación de Datos a Enviar – Curso .....	19
Obtención de Token .....	19
Configuración de Código de curso en la plataforma .....	20
Identificación de Datos a Enviar – Alumno .....	23
Registro del RUT .....	23
Valor de Estado .....	24
Fecha de Inicio y Finalización .....	26
Identificación de Datos a Enviar – Módulo .....	28
Calificación por Módulo .....	30
Identificación de Datos a Enviar – Actividad .....	32
Agregar Bloque de Dedicación al Curso .....	32
Agregar Bloque de Estado de Finalización .....	35
Configurar "Finalización de Actividad" .....	36

Configuración de BigBlueButton para Realización de Clase Sincrónica .....	38
Recomendaciones para Desarrolladores.....	40
Lógicas de Validación dentro del Servicio .....	40
Historial de Avance Enviado .....	40
Manejo de Timeouts y Rendimiento.....	44
Programación Automatizada con Cron.....	45
Cálculo del Tiempo de Dedicación .....	46
Abstracción del modelo de datos: .....	47
Anexos.....	48
Estructurar y Agrupar en Formato JSON .....	48
Abstracción .....	54

## Últimos Cambios Realizados al Instructivo

---

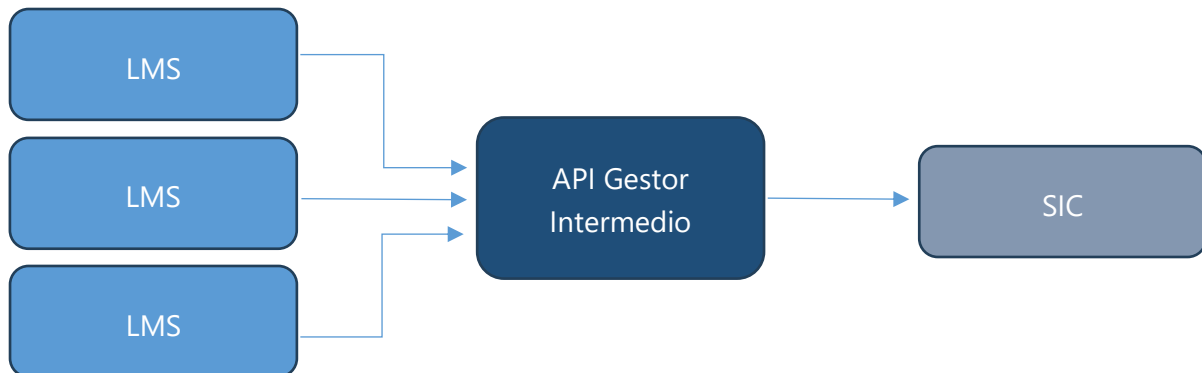
Se ha actualizado y reestructurado este instructivo para incluir la configuración necesaria en Moodle, y recomendaciones para Desarrolladores, con el fin de garantizar el envío correcto de los parámetros.

Se han agregado paso a paso las configuraciones de Moodle para poder obtener los datos correctamente.

1. Agregar bloque de Dedicación al Curso.
2. Agregar bloque de Estado de Finalización.
3. Configurar "Finalización de Actividad".
4. Configurar "Restricciones de Acceso".
5. Configurar las restricciones de acceso a cada módulo por fecha.
6. Configuración de BigBlueButton para realización de clase sincrónica
7. Apartado de preguntas Frecuentes.
8. Recomendaciones para Desarrolladores.

## Introducción

A continuación, se detallan los pasos a seguir para poder realizar el envío de datos desde una LMS<sup>1</sup> Externa a SIC<sup>2</sup> por medio de la API REST del Gestor Intermedio<sup>3</sup>. Esto, con la finalidad de que los ejecutores y los OTECs puedan enviar de forma periódica la información de sus cursos por un canal centralizado.



Para este propósito, se deben seguir los siguientes pasos descritos detalladamente en este documento:

1. Identificación de datos a enviar
2. Estructurar información
3. Obtener autenticación de "Inicio Sesión SENCE"
4. Prueba de conexión
5. Envío de datos a API REST
6. Validación de datos enviados

Además, se recomienda que todo el proceso descrito en este documento se realice como una integración automática y no como un proceso manual de envío de datos por la complejidad que esta puede conllevar. Por lo anterior, se recomienda que este documento sea analizado por un profesional en el área de informática (Ing. Informático, Técnico Informático, Desarrollador, etc.) para su implementación desde el LMS.

<sup>1</sup> Se refiere al concepto de Learning Management System o Sistema de gestión del aprendizaje.

<sup>2</sup> Sistema Integrado de Capacitación de SENCE.

<sup>3</sup> Middleware encargado del envío desde Aula Digital y LMS Externas hacia SIC.

## Identificación de Datos a Enviar

Para comenzar con el proceso de envío de datos desde un LMS a SIC, primero se deben identificar los datos contenidos en el LMS, según las tablas adjuntas más adelante. Esto, con la finalidad de tener la completitud de los datos solicitados por la API antes de poder enviarla.

Curso				
Nombre Parámetro API	Descripción	Tipo Dato	Formato	Observación
codigoOferta	Código del curso	Texto		
codigoGrupo	Código de la sección	Texto		
codigoEnvio	Código de uso interno ejecutor			ID que puede usar cada LMS de forma personal para poder identificar sus transacciones de envío de datos. Es un campo opcional
cantActividadSincronica	Número total de actividades sincrónicas del curso	Número	0 a n	Se valida que sea mayor o igual a 0
cantActividadAsincronica	Número total de actividades asincrónicas del curso	Número	0 a n	Se valida que sea mayor o igual a 0
listaAlumnos	Lista de información de alumnos	Alumno[]		Se valida que no sea una lista vacía

Alumno				
Nombre ParámetroAPI	Descripción	Tipo Dato	Formato	Observación
rutAlumno	RUT del alumno sin dígito verificador	Número		Se valida formato de RUT
dvAlumno	Dígito verificador del alumno	Texto		Se valida si coincide con formato de RUT
tiempoConectividad	Tiempo de conectividad del alumno en segundos	Número		Se valida que sea mayor a 0
estado	Estado del alumno en el curso. Por defecto es 1	Número	1 - 2 - 3	1 - Cursando 2 - Aprobado 3 - Reprobado
porcentajeAvance	Porcentaje avance general	Número Entero	0 - 100	Porcentaje de actividades obligatorias completadas en el curso
fechaInicio	Fecha de inicio del curso	Fecha	YYYY-MM-DD	
fechaFin	Fecha de término del curso	Fecha	YYYY-MM-DD	Se valida que la fecha de fin no pueda ser menor a la de inicio
fechaEjecucion	Representa la fecha a la que corresponde el avance, por defecto la fecha actual	Fecha	YYYY-MM-DD	Se valida que la fecha de ejecución esté entre la fecha de inicio y fin
evaluacionFinal	Representa el porcentaje de nota final del alumno en el curso	Número	0 a 100	Se valida que sea mayor o igual a 0 y menor o igual a 100
listaModulos	Lista de los módulos del alumno	Módulo[]		Se valida que no sea una lista vacía

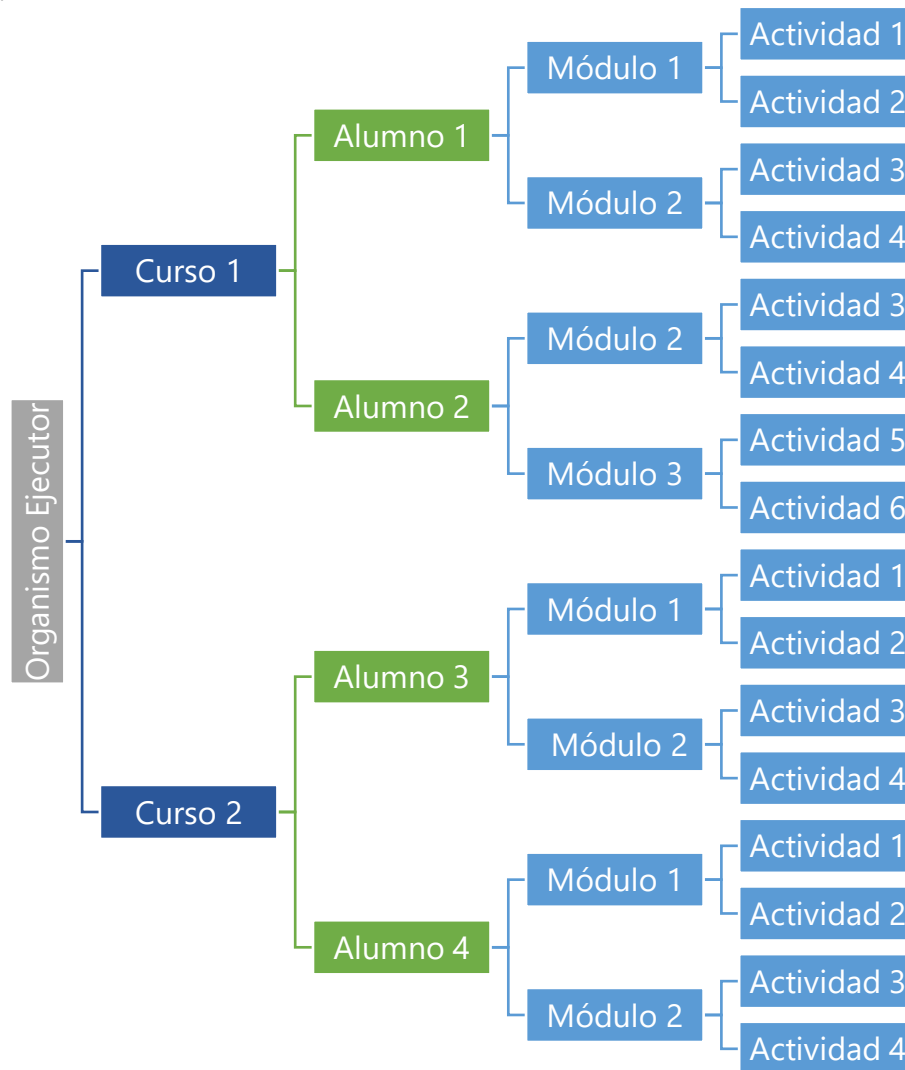
Módulo				
Nombre Parámetro API	Descripción	Tipo Dato	Formato	Observación
codigoModulo	Código del módulo asociado a un alumno	Texto		
tiempoConectividad	Tiempo en segundos	Número		Se valida que sea mayor a 0
estado	Si han realizado todas las actividades, con nota menor a 4 el estado es <b>3</b> , con nota mayor o igual a 4 el estado es <b>2</b> , en cualquier otro caso es <b>1</b>	Número	1, 2 o 3	
porcentajeAvance	Porcentaje de avance del módulo	Número	0 – 100	Porcentaje de actividades obligatorias completadas en el módulo
fechaInicio	Fecha de inicio del curso	Fecha	YYYY-MM-DD	
fechaFin	Fecha de término del curso	Fecha	YYYY-MM-DD	Se valida que la fecha de fin no pueda ser menor a la de inicio
listaActividades	Lista de las actividades del módulo	Actividad[]		Se valida que no sea una lista vacía
notaModulo	Porcentaje de evaluación por módulo	Número	0 a 100	Se valida que sea mayor o igual a 0 y menor o igual a 100
cantActividadSincronica	Número total de actividades sincrónicas del módulo	Número	0 a n	Se valida que sea mayor o igual a 0
cantActividadAsincronica	Número total de actividades asincrónicas del módulo	Número	0 a n	Se valida que sea mayor o igual a 0

Actividad				
Nombre Parámetro API	Descripción	TipoDato	Formato	Observación
codigoActividad	Código de una actividad asociada a un módulo (tareas, encuestas, videos, docs, etc.)	Texto		Si se envía un texto con más de 50 caracteres, este se trunca



## Estructura Árbol Datos

Es importante considerar que un Organismo Ejecutor puede tener más de un programa a ejecutar simultáneamente, en tal situación, se sugiere enviar la información de un curso por cada llamada a la API de SENCE, e ir iterando de esta forma para cada uno de los cursos a reportar. La estructura de la información a enviar se puede abstraer en un árbol como se presenta en el esquema a continuación.



*Esquema de datos*

## Estructurar la Información

Una vez identificada la fuente de los datos mencionados. Se debe estructurar y agrupar en formato JSON para poder enviarlo a la API con los nombres mencionados en las tablas del punto anterior.

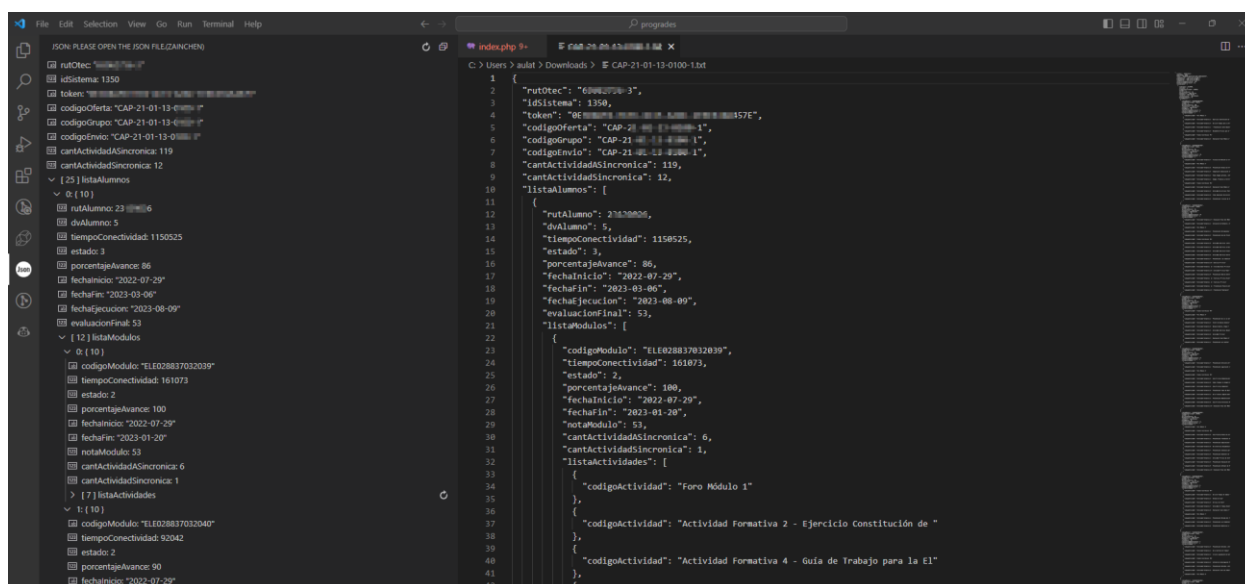
**Ejemplo:** Estructurar y Agrupar en Formato JSON

### Recomendaciones de IDE (Entorno de Desarrollo Integrado) y Extensiones

Al utilizar, por ejemplo, Visual Studio Code junto con la extensión "vscode-json", los desarrolladores pueden asegurarse de que la estructura de sus datos JSON sea precisa antes de enviarlos a la API. Esta extensión ofrece una vista jerárquica de los datos, lo que permite detectar posibles errores de formato o campos faltantes de manera eficiente. Por ejemplo, al trabajar con un JSON que contiene detalles de un estudiante en un curso, se podrá emplear la extensión para navegar por los campos como la dedicación global y por módulo, la nota del curso y por módulo, etc. Esto facilita la identificación rápida de cualquier discrepancia entre la estructura esperada y la proporcionada.

Referencias:

- [Extensión vscode-json](#)



## Obtener Autenticación de “Inicio Sesión SENCE”

---

Con el objetivo de poder identificar a los OTECs vigente que está realizando la integración a SIC, se debe reutilizar el Token que se utiliza en el inicio de sesión SENCE o llave de autenticación de RCE<sup>4</sup>.

**Ejemplo de formato de token:** 5EEBF607-25A9-4DB2-A4DD-5D31BDAE3220

Una vez identificado el token de autenticación se debe agregar a la estructura JSON mencionada en el punto “Identificación de Datos a Enviar – Curso”, los campos de RUT OTEC, ID del Sistema (siempre se envía el valor 1350) y el token como tal.

**Ejemplo:** Estructurar y Agrupar en Formato JSON

---

<sup>4</sup> RCE. Remote Code Execution / Ejecución Remota de Código

## Prueba de Conexión

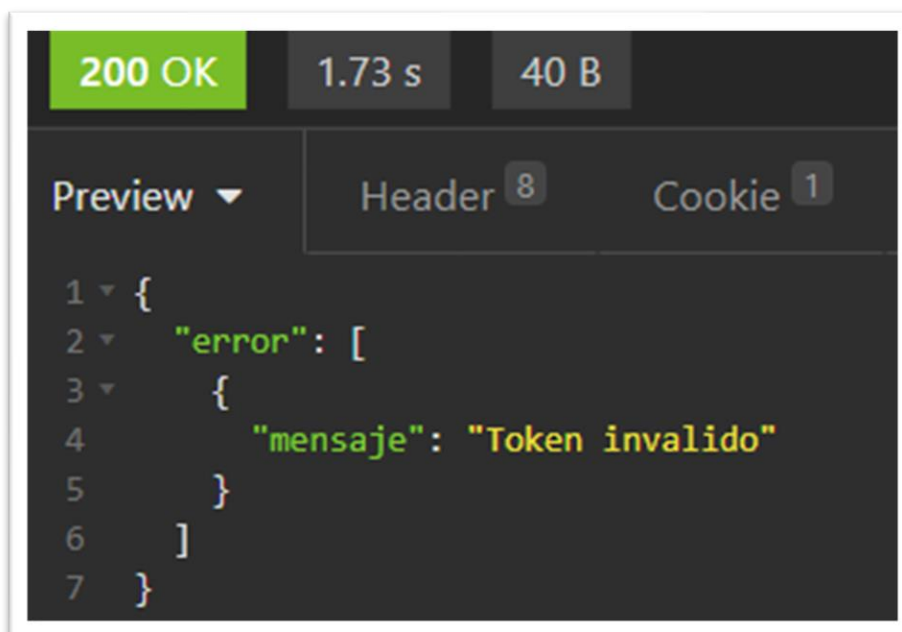
Antes de enviar los datos al servicio, es recomendable validar que se tiene acceso al servidor de la API REST y que este responde correctamente (para descartar restricciones de firewall, error de tipo, etc.). Para ello, se debe hacer la siguiente llamada:

### Método GET

#### End-Point:

```
https://auladigital.sence.cl/gestor/API/avance-sic/historialEnvios?rutOtec=11222333-4&i  
dSistema=1350&token=5EE5EEEE-5555-BDBD-D4D4-E32E32E32E32
```

La cual, debe responder un mensaje de error con código 200 de respuesta como se puede ver en el siguiente ejemplo de un Cliente API REST:



```
200 OK 1.73 s 40 B  
Preview ▾ Header 8 Cookie 1  
1 {  
2   "error": [  
3     {  
4       "mensaje": "Token invalido"  
5     }  
6   ]  
7 }
```

Es decir, se realizó una conexión correcta al servicio, pero con un error.

## Envío de Datos a API REST

---

Con el formato JSON ya estructurado, se debe enviar la información a la API REST según la definición del servicio indicado a continuación:

<b>Método POST</b>
<b>End-Point:</b> <code>https://auladigital.sence.cl/gestor/API/avance-sic/enviarAvance</code>
<b>HEADERS Content-Type:</b> <code>application/json</code>
<b>BODY raw:</b> Estructura JSON completa generada en el punto anterior

Si la información se envía de forma correcta, entonces el servicio lo va a indicar con la siguiente respuesta:

```
{
  "id_proceso": 25,
  "envio": [
    {
      ..estructura de datos enviados
    }
  ],
  "errores": [],
  "respuesta_SIC": "El proceso 25 ha finalizado correctamente, para ver la cantidad de registros y detalle, diríjase a la pantalla de procesos"
}
```

En caso contrario el servicio responderá los datos en el campo "datosError" que se deben revisar al contener algún tipo de problema como se puede ver a continuación:

```
{
  "id_proceso": 31,
  "datosEnviados": [],
  "datosError": [
    {
      "alumno": {
        "rutAlumno": 11111111,
        "dvAlumno": "5",
        "tiempoConectividad": 6000,
        "porcentajeAvance": 20,
        "estado": 1,
        "fechaInicio": "2021-04-01 00:00:00",
        "fechaFin": "2021-08-01 00:00:00",
        "listaModulos": [
          {
            "codigoModulo": "C51737-O14-M1",
            "tiempoConectividad": 600,
            "porcentajeAvance": 50,
            "estado": 1,
            "fechaInicio": "2021-04-01 00:00:00",
            "fechaFin": "2021-08-01 00:00:00",
            "listaActividades": [
              {
                "codigoActividad": "Tarea de prueba"
              }
            ]
          }
        ]
      },
      "codigo": "012",
      "mensaje": "El alumno 11111111-5 no se encuentra registrado."
    }
  ],
  "respuesta_SIC": ""
}
```

De esta manera, se presentan a continuación los códigos de error asociados al servicio en caso de que se produzcan problemas con la estructura o el contenido de los datos enviados. Estos códigos se detallan en la siguiente tabla:

Código Error	Texto Error	Descripción
001	Token inválido	Alguno de los datos especificados para la autenticación (idSistema, RUT, token) no son válidos, cuando se produce este error no se guarda registro del avance enviado.
002	Descripción del error ocurrido	Se produce cuando hay algún error con la autenticación, puede ser desde problemas de conexión hasta problemas del servicio de autenticación de SENCE.
003	El curso (codigoCurso) no se encuentra registrado	Se produce cuando el código de curso enviado no existe en el Gestor Intermedio
010	El alumno (rutAlumno) no se encuentra registrado en el módulo (codigo Módulo)	El alumno si está registrado en la plataforma, pero no pertenece al módulo.
011	El alumno (rutAlumno) no se encuentra registrado en este curso.	El alumno si está registrado en la plataforma, pero no pertenece al curso.
012	El alumno (rutAlumno) no se encuentra registrado.	El alumno no está registrado en la plataforma de SENCE, este error solo afecta a este alumno, los alumnos correctos si serán procesados.
021	alumno.tiempoConectividad inválido	El tiempo de conectividad indicado para el alumno no es numérico o es menor a 0
022	alumno.porcentajeAvance inválido	El porcentaje de avance indicado para el alumno no es numérico o es menor a 0
023	modulo.tiempoConectividad inválido	El tiempo de conectividad indicado para el módulo no es numérico o es menor a 0
024	modulo.porcentajeAvance inválido	El porcentaje de avance indicado para el módulo no es numérico o es menor a 0
025	modulo.porcentajeAvance debe ser mayor al anterior (porcentajeAnterior)	Cuando se envía un porcentaje de avance menor al ya ingresado previamente.
026	modulo.fechalnicio inválido	Cuando se envía una fecha de inicio del módulo que no tiene formato "yyyy-mm-dd"

Código Error	Texto Error	Descripción
027	modulo.fechaFin inválido	Cuando se envía una fecha de fin del módulo que no tiene formato "yyyy-mm-dd"
028	modulo.notaModulo inválido	El porcentaje de nota indicado para el módulo no es número o es menor a 0
029	modulo.cantActividadSincronica inválido	La cantidad de actividades sincrónicas indicadas para el módulo no es número o es menor a 0.
030	modulo.cantActividadAsincronica inválido	La cantidad de actividades asincrónicas indicadas para el Módulo no es número o es menor a 0.
031	alumno.evaluacionFinal inválido	La evaluación final indicada para el curso no es número o es menor a 0.
032	curso.cantActividadSincronica inválido	La cantidad de actividades sincrónicas indicadas para el módulo no es número es menor a 0.
033	curso.cantActividadAsincronica inválido	La cantidad de actividades asincrónicas indicadas para el Módulo no es número o es menor a 0.



## Preguntas Frecuentes

---

1. ¿Cada vez que tenga que enviar la información se debe cumplir con los pasos indicados?

No, esta secuencia de pasos es para comprender el funcionamiento de la API, pero es recomendable automatizar el envío de datos a API REST teniendo como input lo realizado en los puntos anteriores.

2. ¿Se puede revisar la historia de los datos que se han enviado?

Sí, existe un método get adicional (especificado en Prueba de Conexión, página 12), para poder ver el historial de todas las interacciones que se han tenido con la API.

3. ¿Puedo usar un sistema distinto a Moodle para poder integrar a la API?

Sí, se puede utilizar cualquier sistema, siempre y cuando, se tenga la totalidad de los datos necesarios.

4. Si se falla solamente el envío de algunos datos, ¿Se debe enviar todo nuevamente?

No es necesario, la API carga los datos correctamente enviados e indica por qué fallaron los datos incorrectos para su corrección y envío posterior de forma independiente.

5. ¿Podré ver la información en SIC inmediatamente después de enviar los datos?

No, al enviar el avance desde las LMS estará visible en SIC al día siguiente, es un proceso nocturno que se realiza desde las 01:00 hasta las 06:00 todos los días.

6. ¿Se actualiza la información de un alumno si se envía más de una vez?

Sí, cada vez que se envían los datos, se actualizarán y sobrescribirá la información anterior. Además, una vez enviado el % avance, la próxima vez que se envíen debe ser mayor al valor previamente enviado.

7. ¿Cuál debería ser la cantidad de datos y frecuencia con que se debe enviar la información a la API?

Se debe enviar la información solo una vez al día en el horario establecido, esto es entre las 22:00 y las 00:00 hrs.

**8.** ¿Se pueden enviar varios datos a la vez?

Sí, se puede enviar una lista de alumnos, como también, se puede enviar solo un alumno. De esta manera, la API responderá identificando cual es el dato que tiene el problema en caso de existir.

**9.** ¿Cómo canalizar las consultas que nos puedan ir saliendo al realizar la API?

Canalizar por medio de correo [adminlearning@sence.cl](mailto:adminlearning@sence.cl)

**10.** ¿Cuál es el alcance de esta API?

Esta API refiere a la conexión de sistemas para la lectura de datos de avance. Distinto es integración de asistencia y botón de acceso, estos se mantienen tal cual está actualmente.

**11.** ¿Cuál es el formato del token correcto?

Todo el token debe ser con caracteres en mayúscula.

## Configuración de Moodle

---

La configuración precisa y adecuada de su LMS es un paso crucial para garantizar el envío correcto de los parámetros. Es especialmente importante asegurarse, por ejemplo, de lo siguiente:

1. Cada módulo debe contar con al menos una clase sincrónica configurada utilizando BigBlueButton. Si se utiliza otra plataforma tiene que ser integrada como "Plugin" para la conexión de las clases (Como ZOOM, Teams, Google Meet u otra), deben quedar debidamente incrustada en la plataforma ya que de lo contrario no podremos garantizar que los tiempos de conexión de estas clases se sumen al total de conectividad.
2. El nombre de los módulos tiene que estar escrito utilizando el código correspondiente entregado por SIC.
3. Los alumnos deben tener configurada la fecha de inicio y término de matrícula.

A continuación, se proporcionará la configuración correcta de la plataforma LMS, para este ejemplo utilizaremos Moodle 4.0, este paso a paso estará realizado en función de los parámetros solicitados por la API REST del Gestor Intermedio .

## Identificación de Datos a Enviar – Curso

### Obtención de Token

El Token es una subclave que permite a los organismos ejecutores dar acceso a desarrolladores externos para la generación de plugins sin comprometer el acceso a la información operativa de la misma. Se trata de un Identificador Único Universal (UUID, por sus siglas en inglés). Los UUID son identificadores de 128 bits que se muestran típicamente en un formato compuesto por grupos de dígitos hexadecimales separados por guiones. Se utilizan para identificar de manera única recursos u objetos en varios sistemas y aplicaciones.

Para obtenerlo, el organismo debe ingresar a <https://sistemas.sence.cl/rts> con el RUT del Organismo Ejecutor y el RUT de la Representante Legal registrado frente a SENCE, junto a su CUS. Una vez dentro deberá hacer clic en "Generar nuevo Token" y copiar la cadena alfanumérica que arrojará el sistema. Es importante indicar que el Token debe indicarse en mayúsculas en los datos formateados como JSON que se enviarán al endpoint del Servicio.



### Administrador Token SENCE

Bienvenido(a): **LUIS RAMIREZ LOPEZ** S.  
OTE

[Salir](#)

## Token

Mostrar **10** registros      Buscar:

Rut Empresa	Rut Representante Legal	Token	Fecha de Registro	Fecha Caducidad
11111111-4	4.11111111-0	571FAF11-11111111-1111-1111-D68559BCC94D	15-05-2020	Token sin caducidad <input type="checkbox"/>

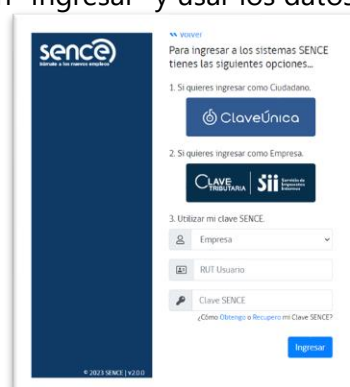
Mostrando registros del 1 al 1 de un total de 1 registros      Anterior **1** Siguiente

[Generar Nuevo Token](#)   [Eliminar Token Seleccionado](#)

## Configuración de Código de curso en la plataforma.

Estando dentro de la plataforma, ingresar al curso a "Configuración", aquí se encuentra la opción de "Nombre corto del curso" en este espacio ingresar el "Código del curso" entregado por SIC.

Para poder obtener el código de curso en caso de no tenerlo, debe ingresar a <https://sicadministrativo.sence.cl/>, presionar en "Ingresar" y usar los datos de su ClaveÚnica.



Al ingresar, seleccionar la pestaña "MANTENEDORES" y la opción "CURSOS"



Se despliega esta ventana donde debe seleccionar el programa al que pertenece el curso para poder ver el código de este, también puede seleccionar por año.

Mantenedor de Cursos

---

Mantenedor de Cursos de Capacitación

Programa:	<input type="text" value="Seleccione Programa"/>	Número Llamado:	<input type="text" value="Seleccione Llamado"/>
Región:	<input type="text" value="Todas"/>	Comuna:	<input type="text" value="Todas"/>
Código Curso:	<input type="text"/>	Código Oferta:	<input type="text"/>
Año Presupuestario:	<input type="text" value="2021"/>	OTEC :	<input type="text" value=""/>
Entidad Requirente :	<input type="text" value="Seleccione Entidad Requirente"/>	OTIC :	<input type="text" value="Seleccione OTIC"/>
Institución Pública :	<input type="text" value="Seleccione Institución Pública"/>	Población Objetivo:	<input type="text" value="TODOS"/>
Estado Curso :	<input type="text" value="Todos"/>		

Luego de seleccionar el programa, presionar el botón "Buscar", con esto se despliega el listado de todos los cursos que pertenecen a este programa. Aquí se puede obtener el código del curso que se está configurando.

Cursos de Capacitación

Exportar a Excel

Acción	Info OTEC Validada	Código Curso ^	Modalidad Instruccional	Nombre Curso	Código Oferta	Program	OTEC	Comuna	Estado
	-	BFCS-21-02-14-01	Presencial (Pasa a	COMERCI	274714101	PROGRA	CIAPE M.	VALDIVI	Fase Int
	-	BFCS-21-02-13-00	Presencial (Pasa a	COMERCI	274713130	PROGRA	CIAPE M.	LO ESPE	Cerrado
	-	BFCS-21-02-13-00	Presencial (Pasa a	COMERCI	274713110	PROGRA	CIAPE M.	SANTIAC	Cerrado
	-	BFCS-21-02-05-00	Presencial (Pasa a	HERRAMIE	27385502	PROGRA	CIAPE M.	VILLA AL	Fase Int
	-	BFCS-21-02-05-00	Presencial (Pasa a	HERRAMIE	27385804	PROGRA	CIAPE M.	CABILDC	Deshabil
	-	DMIPE-21-02-05-0	Presencial (Pasa a	FORMULA	20055404	BONO DI	CIAPE M.	PETORCA	Deshabil

Página 1 de 2    Mostrando 1 - 10 de 12

## Identificación de Datos a Enviar – Alumno

---

En el caso del alumno hay que realizar 2 configuraciones que son:

1. Registro del RUT
2. Valor de estado
3. Fecha de inicio y fin del curso

### Registro del RUT

Para la configuración del RUT del alumno se debe ingresar a "Participantes", seleccionar al participante, "Editar perfil", al final está la categoría "Opcional", dentro encontrarás el "Número de ID" aquí debes ingresar el RUT sin puntos y con guion.

> Intereses

▼ Opcional

Número de ID	<input type="text" value="16111222-3"/>
Institución	<input type="text"/>
Departamento	<input type="text"/>



## Valor de Estado

En la estructura de los datos a enviar a SENCE se debe incluir un valor llamado "Estado" tanto a nivel de curso como por cada uno de los módulos del mismo. Para esto será necesario setear a nivel de plataforma los criterios de aprobación que deben lograr los participantes de los distintos programas. Se sugiere una página de configuración como la que se presenta a continuación.

### SIC SENCE

**Requisitos de Aprobación**

Porcentaje Progreso para Aprobar  Valor por defecto: 99  
local\_sicence | porcentajeaprobacion

---

Calificación Mínima para Aprobar  Valor por defecto: 59  
local\_sicence | notaaprobacion

---

**Ajuste Cálculo Dedicación**

Minutos Máximo entre Clics  Valor por defecto: 150  
local\_sicence | clicspan

---

**Endpoint**

Endpoint Envío Datos  Valor por defecto: https://srvinterop.sence.cl/wsconnel/api/indicadores/setindicadoresExternos  
local\_sicence | endpointsence

Con estos datos, se puede establecer una lógica como la que proponemos a continuación a modo de ejemplo:

```

if ($notacurso > $notaaprobacion && $progresogeneral >
$porcentajeaprobacion) {
    $estado = 2; // Aprobado
} else if ($matriculatimeend < time()) {
    $estado = 3; // Reprobado
} else {
    $estado = 1; // Cursando
}
    
```

En el ejemplo de código anterior se utilizan condiciones para determinar el estado de un estudiante en un curso, considerando varios factores como la nota obtenida en el curso, el progreso general y el tiempo de finalización de la matrícula.

- **Si la nota del curso es mayor que la nota de aprobación y el progreso general es mayor que el porcentaje de aprobación**, entonces el estudiante ha aprobado el curso. Por lo tanto, se establece el valor del estado en 2, que representa "aprobado".
- **Si el tiempo de finalización de la matrícula ha pasado**, lo que se compara con el tiempo actual (obtenido mediante `time()`), entonces el estudiante ha reprobado el curso debido a que no completó el curso a tiempo. En este caso, se establece el valor del estado en 3, que representa "reprobado".
- **Si ninguna de las condiciones anteriores se cumple**, significa que el estudiante está en proceso de cursar el programa. Por lo tanto, se establece el valor del estado en 1, que representa "**cursando**".

Esta lógica evalúa las condiciones necesarias para determinar el estado actual del estudiante en función de su rendimiento, progreso y cumplimiento de plazos.

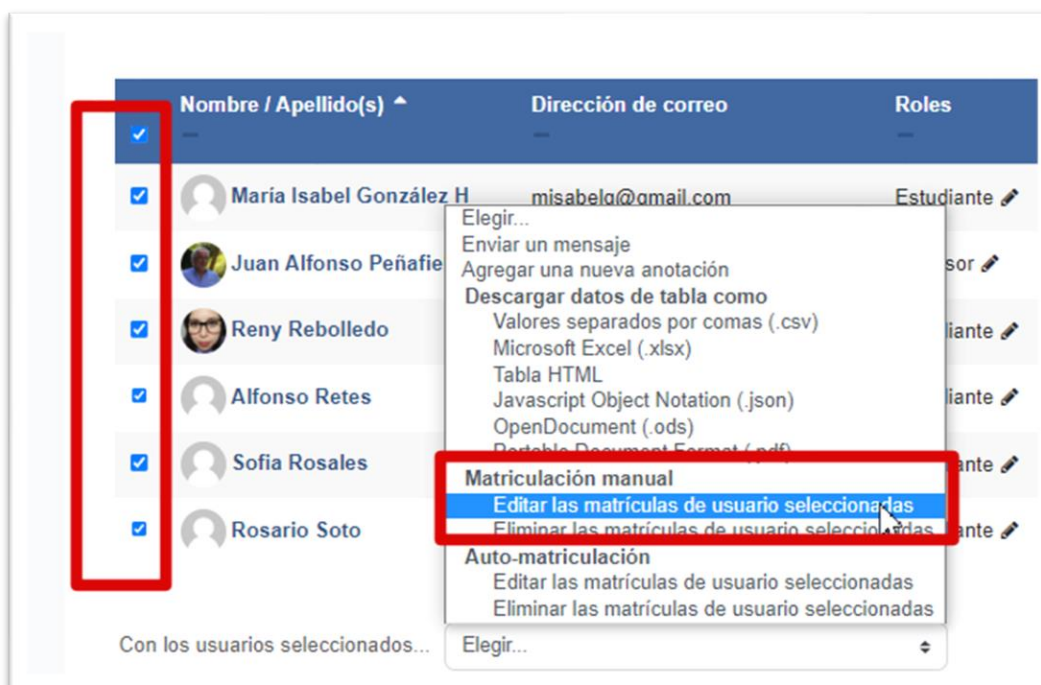
## Fecha de Inicio y Finalización

Uno de los datos requeridos a ser enviado al Gestor Intermedio son las fechas de inicio, término y de ejecución para cada usuario en cada programa. Acá nos enfrentaremos a dos situaciones. Según se observa en la sección de "Configuración de plataforma", se sugiere que el organismo ejecutor configure el inicio y término de la matrícula para cada usuario. Es posible que esto no sea necesario -o sea omitido- por lo que es necesario tener fallbacks dispuestos. El caso en que no es necesario es cuando todos los participantes del curso participan del mismo programa. Si hay cohortes al interior del curso, será necesario explicitar inicio y fin de matrícula para cada una de ellas. Una idea es obtener estas fechas desde los datos del curso. De esta forma podríamos usar un fallback como el que sigue:

```
//obtiene fechas de inicio y fin de matrícula  
  
$matricula = sicsence_fechamatricula($curso->id, $user->id);  
  
$matriculatimestart = $matricula->timestart ?? $curso->startdate;  
  
$matriculatimeend = $matricula->timeend ?? $curso->enddate;  
  
$ejecuciontime = date('Y-m-d', time());
```

En el caso de Moodle los datos de matrícula están almacenados en la tabla {user\_enrolments} y los datos del curso en la tabla {course}. En último caso, indicar algún error en la gestión del sitio, ya que podríamos estar enviando un dato NULL hacia el SIC lo que tendrá como consecuencia un error en el registro del módulo o del curso del usuario.

En la plataforma se debe ingresar "Participantes" seleccionamos todos los alumnos y en la parte inferior buscamos la opción "Editar las matrículas de usuarios seleccionadas", haciendo clic nos mostrará otra ventana.



Nos mostrará la opción de cambiar el tiempo inicial y final, seleccionamos el cuadro de "Habilitar" y ponemos las fechas que correspondan.

Cambiar estado: Sin cambios

Cambiar tiempo inicial: 1 julio 2023 23 50  Habilitar

Cambiar tiempo final: 1 julio 2023 23 50  Habilitar

**GUARDAR CAMBIOS** **CANCELAR**

Cambiar estado: Sin cambios

Cambiar tiempo inicial: 1 julio 2023 23 50  Habilitar

Cambiar tiempo final: 20 julio 2023 23 50  Habilitar

**GUARDAR CAMBIOS** **CANCELAR**

## Identificación de Datos a Enviar – Módulo

Para obtener los códigos de cada Módulo, debe ingresar a <https://sicadministrativo.sence.cl/> con su Clave Única, ingresar en la pestaña "MANTENEDORES" y seleccionar "CURSOS"

En este Mantenedor de Cursos ingresar el código del curso que se está configurando y presionar el botón "Buscar"

En la parte inferior de la pantalla se desplegará la información del curso, debe ingresar en el icono " + "

### Mantenedor de Cursos de Capacitación

Programa:     Número Llamado:

Región:     Comuna:

Código Curso:     Código Oferta:

Año Presupuestario:     OTEC :

Entidad Requirente :     OTIC :

Institución Pública :     Población Objetivo:

Estado Curso :

---

### Cursos de Capacitación

Exportar a Excel

Acción	Info OTEC Validada	Código Curso ^	Modalidad Instruccional	Nombre Curso	Código Oferta	Program	OTEC	Comuna	Estado
	-	BFCS-21-02-14-01	Presencial (Pasa a	COMERCI	274714101	PROGRA	CIAPE M	VALDIVI	Fase Int

Se abrirá una ventana con el código y nombre de Módulo.

### Mantenedor de Módulos Curso

Modulos Curso BFCS-21-02-14-0106-3

Estos módulos estarán disponibles para generar horario y asignar relator E-LEARNING (EXCEPTO MÓDULOS NO EXIGIBLES).

Acción	Código	Nombre	Horas	Modalidad Instruccional	Exigible
	PRES028008000558	INTRODUCCIÓN AL COMERCIO ELECTRÓNICO	10,00	Elearning	Si
	PRES028008000559	COMERCIO ELECTRÓNICO Y COMUNICACIÓN DIGITAL	30,00	Elearning	Si
	PRES028008000560	HERRAMIENTAS DE MARKETING DIGITAL	30,00	Elearning	Si
	PRES028008000561	SEGURIDAD DIGITAL Y ATENCIÓN AL CLIENTE EN COMERCIOS ELECTRONICOS	20,00	Elearning	Si

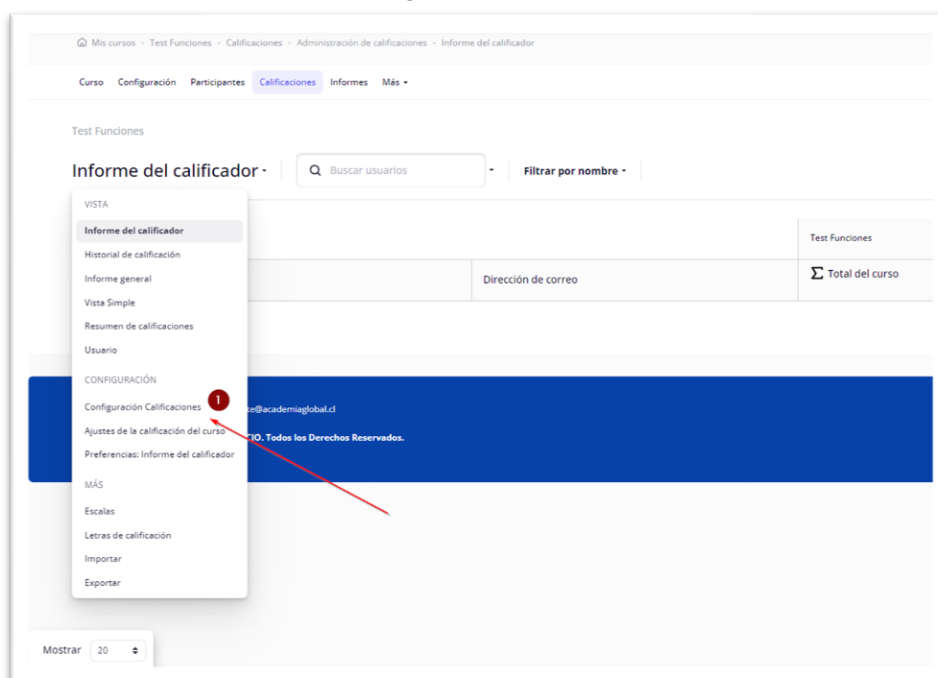
Con esta información puede configurar cada uno de los Módulos del curso ingresando estos datos como el siguiente ejemplo:



## Calificación por Módulo

En la actualización requerida por el Gestor Intermedio, se incluye el requerimiento de informar la calificación del participante por cada módulo del programa. Para ordenar estos datos desde el punto de vista de la gestión de la plataforma, será pertinente que agrupemos las distintas calificaciones de cada módulo en el gestor de calificaciones del LMS en uso.

De esta forma, por ejemplo, en el caso de Moodle, deberemos crear categorías de calificaciones al interior del curso, con el nombre de los distintos Módulos del programa, de la forma en que se presenta en las siguientes capturas:



Test Funciones

Configuración Calificaciones · Añadir ítem de calificación Añadir categoría

Nombre	Ponderaciones	Calificación máxima	Estado
Test Funciones			Suma Excluir calificaciones vacías
CÁLCULO TOTAL			
Total del curso		0.00	

Guardar cambios

Categoría de calificación

Nombre de la categoría \*

Cálculo total

Ver menos...

Excluir calificaciones vacías !

Descartar las más bajas

Total categoría

Mostrar más...

Tipo de calificación

Escala

Calif. máxima

Calif. mínima

Ocultar

Guardar cambios Cancelar

Curso Configuración Participantes **Calificaciones** Informes Más ▾

Test Funciones

Configuración Calificaciones · Añadir ítem de calificación Añadir categoría

Nombre	Calificación máxima	Estado	Acciones	Seleccionar
Test Funciones		Media ponderada simple de calificaciones	Excluir calificaciones vacías	<input type="checkbox"/> Todos
ELE028837032039		Media ponderada simple de calificaciones		<input type="checkbox"/> ELE028837032039
CÁLCULO TOTAL				
Total ELE028837032039	100.00			
ELE028837032040		Media ponderada simple de calificaciones		<input type="checkbox"/> ELE028837032040
CÁLCULO TOTAL				
Total ELE028837032040	100.00			
CÁLCULO TOTAL				
Total del curso	100.00			

Guardar cambios

Mover los ítems seleccionados a

Modo de edición



## Identificación de Datos a Enviar – Actividad

Aquí detallaremos las actividades que se deben configurar con algunas especificaciones muy concretas (puede encontrar más información de como configurar actividades en el “Manual para Ejecutores”):

Los bloques “dedicación al Curso” y “Estado de finalización” son plugins que se deben descargar desde <https://moodle.org/plugins> e instalar en la plataforma antes de agregarlos al curso.

1. Agregar bloque de Dedicación al Curso.
2. Agregar bloque de Estado de Finalización.
3. Configurar “Finalización de Actividad”.
4. Configurar las restricciones de acceso a cada módulo por fecha.
5. Configuración de BigBlueButton para realización de clase sincrónica.

### Agregar Bloque de Dedicación al Curso.

Al ingresar al curso en la parte superior derecha se encuentra una flecha, al presionarlo se despliega “caja del bloque”



En esta "caja del bloque" se pueden ver todos los bloques del curso, en la parte superior se encuentra la opción de "AGREGAR UN BLOQUE"

The screenshot displays the SENCE course management interface. At the top, there are navigation tabs: "Curso", "Configuración", "Participantes", "Calificaciones", "Informes", and "Más". A red box highlights the "+ AGREGAR UN BLOQUE" button in the top right corner. Below the tabs, the course title "FORTALECIENDO MI PROPUESTA DE VALOR / C48006-015387-M1" is shown. A list of blocks is displayed, including "FORO Avisos", "BIGBLUEBUTTON Case N1", "RECURSO Material de apoyo", "URL Video explicativo", and "CUESTIONARIO Cuestionario N1". A status bar at the bottom indicates "No disponible hasta que: La actividad Video explicativo esté marcada como realizada". On the right sidebar, the "Estado De Finalización" section shows a progress bar and a "VISTA GENERAL DE ESTUDIANTES" button. The "Integración SENCE" section displays a warning: "¡CONFIGURACIÓN INCOMPLETA! Se debe: • Configurar OTEC del Sitio • Configurar un Código de Curso de 10 dígitos".

Al ingresar se muestran un conjunto de bloques para asignar, se debe seleccionar "Dedicación al Curso"

The screenshot shows a dialog box titled "Agregar un bloque" with a close button (X) in the top right corner. The dialog contains a list of options: "Calendario", "Comentarios", "Cursos", "Dedicación al curso", and "Enlaces de sección". The "Dedicación al curso" option is highlighted with a red rectangular box.

Se mostrará de esta manera:

The screenshot displays a course management interface. The main content area shows a list of activities for the course 'FORTALECIENDO MI PROPUESTA DE VALOR / C48006-015387-M1'. The activities include:

- FORO Avisos
- BIGBLUEBUTTON Case N1 (Por hacer: Ver)
- RECURSO Material de apoyo (Por hacer: Ver)
- URL Video explicativo (Hecho: Ver)
- CUESTIONARIO Cuestionario N1 (Por hacer: Ver)

A lock icon at the bottom indicates: 'No disponible hasta que: La actividad Video explicativo esté marcada como realizada'.

The right sidebar contains several blocks:

- AGREGAR UN BLOQUE
- Dedicación Al Curso** (highlighted with a red box):
  - Sólo para profesores:
  - HERRAMIENTA DE DEDICACIÓN
- Estado De Finalización:
  - AHORA
  - Informes de reuniones: No finalizado
  - VISTA GENERAL DE ESTUDIANTES
- Integración SENCE:
  - ¡CONFIGURACIÓN INCOMPLETA!
  - Se debe:

## Agregar Bloque de Estado de Finalización

Al igual que en el caso anterior, ingresar a "caja del bloque" seleccionar la opción de "AGREGAR UN BLOQUE" y buscar "Estado de Finalización"



Se mostrará de esta manera:

A screenshot of a course management interface. The main content area shows a list of course activities for "FORTALECIENDO MI PROPUESTA DE VALOR / C48006-015387-M1". The right sidebar contains several sections: "AGREGAR UN BLOQUE" (with a red box around the "Estado De Finalización" block), "Dedicación Al Curso", and "Integración SENCE". A notification at the bottom of the sidebar says "¡CONFIGURACIÓN INCOMPLETA!".

## Configurar “Finalización de Actividad”

Agregada una actividad, se debe configurar la “Finalización de actividad” dentro de cada módulo.

Acceder a “Finalización de actividad”, y en la opción de “Rastreo de finalización” seleccionar la opción “Mostrar la actividad como completada cuando se cumplan las condiciones”.

> Apariencia

> Ajustes comunes del módulo

> Restricciones de acceso

▼ Finalización de actividad

Rastreo de finalización ? Los estudiantes pueden marcar manualmente la actividad como completada ⇅

Se espera finalizar en ? 3 julio 2023 08:58  Habilitar

> Marcas

> Competencias

Send content change notification ?

GUARDAR CAMBIOS Y REGRESAR AL CURSO GUARDAR CAMBIOS Y MOSTRAR CANCELAR

▼ Finalización de actividad

Rastreo de finalización ? Los estudiantes pueden marcar manualmente la actividad como completada ⇅

Se espera finalizar en ? No indicar finalización de la actividad

Mostrar la actividad como completada cuando se cumplan las condiciones

En la opción "Requerir ver", se debe seleccionar la opción "El Estudiante debe ver esta actividad para finalizarla". De esta forma, se asegura que el estudiante deba acceder a la actividad y recurso respectivo, para que la plataforma reconozca su avance.

Finalización de actividad

Rastreo de finalización ? Mostrar la actividad como completada cuando se cumplan las condiciones

**Requerir ver**  El estudiante debe ver esta actividad para finalizarla

Se espera finalizar en ? 3 julio 2023 08 58  Habilitar

Si un recurso es configurado como complementario, debe seleccionar la opción "No indicar finalización de la actividad"

Finalización de actividad

Rastreo de finalización ? No indicar finalización de la actividad

*Nota: Todos los módulos deben tener actividades obligatorias.*

Desde la versión de Moodle 4.0 en adelante, los estados de Finalización de actividad se ven de la siguiente manera.

URL  
Video explicativo ✓ Hecho: Ver

CUESTIONARIO  
Cuestionario N1 Por hacer: Ver

No disponible hasta que: La actividad Video explicativo esté marcada como realizada

## Configuración de BigBlueButton para Realización de Clase Sincrónica.

Desde la versión de Moodle 4.0 en adelante, Moodle trae incorporado el plugin de BBB<sup>5</sup>, por esto solo hay que activarlo y usarlo dentro de cada módulo.

Para activar BBB debe ingresar a "Administración del sitio", seleccionar la pestaña "Extensiones", bajar hasta encontrar la sección "Módulos de Actividad" e ingresar en "Gestionar de actividades"



Dentro se encuentran todas las actividades que se pueden utilizar en el curso, como se puede ver en la siguiente imagen BBB se encuentra Oculto.

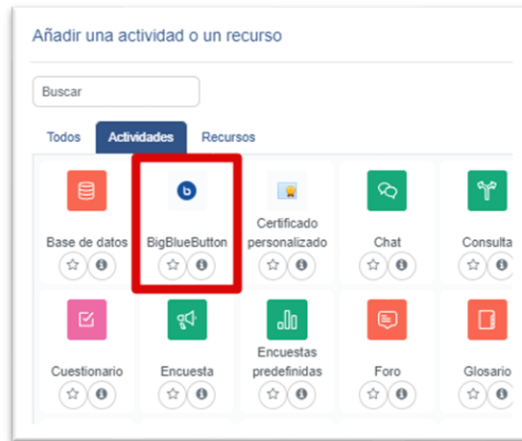
Módulo de Actividad	Actividades	Versión	Ocultar/Mostrar	Configuración	Desinstalar
Tarea	0	2022041900		Configuración	Desinstalar
Tarea 2.2 (Deshabilitado)	0	2022041900			Desinstalar
BigBlueButton	4	202204190		Configuración	Desinstalar
Libro	0	2022041900		Configuración	Desinstalar
Chat	1	2022041900		Configuración	Desinstalar

Se debe realizar un clic sobre el ojo en gris, y se pondrá en azul, esto indica que ya estará disponible para uso.

Módulo de Actividad	Actividades	Versión	Ocultar/Mostrar	Configuración
Tarea	0	2022041900		Configuración
Tarea 2.2 (Deshabilitado)	0	2022041900		
BigBlueButton	4	2022041901		Configuración

<sup>5</sup> BigBlueButton

Teniendo lo anterior listo, ya se podrá seleccionar la actividad "BigBlueButton"



Para mayor detalle de como configurar el completo BBB revisar el "Manual para Ejecutores".



## Recomendaciones para Desarrolladores

---

### Lógicas de Validación dentro del Servicio

Con el objetivo de tener un primer filtro de validación de los datos a nivel de Gestor Intermedio (previo a la llegada a SIC). Están implementadas las siguientes lógicas de negocio dentro del servicio:

1. Que el último % de avance enviado en un día no pueda ser menor al % avance de un día posterior.
2. Validar existencia de RUT del OTEC y código de curso a enviar al Gestor Intermedio.
3. Validar que los datos a enviar (RUT alumno / módulo / curso), coinciden con los registrados en el Gestor Intermedio , evitando que se envíen avances de alumnos no registrados en el GI.

### Historial de Avance Enviado

Se puede revisar el historial de la información enviada por medio de la API utilizando el servicio siguiente:

#### Método GET

---

End-Point:

<https://auladigital.sence.cl/gestor/API/avance-sic/historialEnvios?rutOtec=11222333-4&idSistema=1350&token=5EE5EEEE-5555-BDBD-D4D4-E32E32E32E32>

Parámetros:

Nombre Campo	¿Es Requerido?	Descripción	Ejemplo
<b>rutOTEC</b>	Si	RUT del OTEC sin puntos y con dígito verificador después de un guion	11222333-4
<b>idSistema</b>	Si	ID Sistema para validar token del OTEC	1350
<b>token</b>	Si	Token autenticación del OTEC	5EE5EEEE-5555-BDBD-D 4D4-E32E32E32E32
<b>fechaDesde</b>	No	Fecha desde cuando se obtienen los registros	2021-08-04
<b>codigo_externo</b>	No	Código personalizado nombrado como "codigoEnvío" en el proceso de envío.	TEST-AV-002
<b>id_proceso</b>	No	ID del proceso recibido luego de cada envío exitoso	567

De esta manera, se puede consultar todo el historial asociado al OTEC, o filtrar por los parámetros de fecha, código externo o id de proceso, entregando la siguiente respuesta

```
[
  {
    "id_proceso_externo": "30",
    "id_lms": "0",
    "codigo_externo": "qa-f-2",
    "fecha": "2021-07-27 02:11:13",
    "horario": null,
    "observaciones": "El proceso 30 ha finalizado correctamente, para ver la cantidad de registros y detalle consulte el servicio \vavance-sic\historial-envios?id_proceso=30 ",
    "observaciones_sic": null,
    "status_code": "0",
    "estado": "2",
    "rut_otec": "77124930",
    "listaregistros": [
      {
        "id_registro": "44",
        "rut_otec": "77124930",
        "dv_otec": "2",
        "oferta_cod": "AYSEN-20-03-11-0007",
        "seccion_cod": "C51737-O14-M2",
      }
    ]
  }
]
```

```
"rut_alumno": "9562011",
"dv_alumno": "6",
"tiempo_conectividad": "200",
"estado": "1",
"porcentaje_avance": "10.00000",
"fecha_inicio": "2021-04-01 00:00:00",
"fecha_fin": "2021-08-01 00:00:00",
"fecha_ejecucion": "2021-07-27 20:37:06",
"mod_cod": "C51737-O14-M2",
"mod_tiempo_conectividad": "120",
"mod_estado": "1",
"mod_porcentaje_avance": "10.00000",
"mod_fecha_inicio": "2021-04-01 00:00:00",
"mod_fecha_fin": "2021-08-01 00:00:00",
"mod_obligatorio": null,
"mod_nota": null,
"act_cod": null,
"act_estado": null,
"estado_registro": "3",
"obs_registro": "[010] modulo.porcentajeAvance debe ser mayor al anterior (20.00000)",
"id_proceso_externo": "30"
},
{
  "id_registro": "45",
  "rut_otec": "77124930",
  "dv_otec": "2",
  "oferta_cod": "AYSEN-20-03-11-0007",
  "seccion_cod": "C51737-O14-M1",
  "rut_alumno": "15294646",
  "dv_alumno": "5",
  "tiempo_conectividad": "6000",
  "estado": "1",
  "porcentaje_avance": "20.00000",
  "fecha_inicio": "2021-04-01 00:00:00",
  "fecha_fin": "2021-08-01 00:00:00",
  "fecha_ejecucion": "2021-07-27 20:37:06",
  "mod_cod": "C51737-O14-M1",
  "mod_tiempo_conectividad": "600",
  "mod_estado": "1",
  "mod_porcentaje_avance": "50.00000",
  "mod_fecha_inicio": "2021-04-0 00:00:00",
  "mod_fecha_fin": "2021-08-01 00:00:00",
  "mod_obligatorio": null,
  "mod_nota": null,
  "act_cod": null,
  "act_estado": null,
  "estado_regist": "1"
```

```

"obs_registro": "",
"id_proceso_externo": "30"
},
{
"id_registro": "46",
"rut_otec": "77124930",
"dv_otec": "2",
"oferta_cod": "AYSEN-20-03-11-0007",
"seccion_cod": "C51737-O14-M2",
"rut_alumno": "18500530",
"dv_alumno": "5",
"tiempo_conectividad": "18000",
"estado": "1",
"porcentaje_avance": "55.00000",
"fecha_inicio": "2021-04-01 00:00:00",
"fecha_fin": "2021-08-01 00:00:00",
"fecha_ejecucion": "2021-07-27 20:37:06",
"mod_cod": "C51737-O14-M2",
"mod_tiempo_conectividad": "1800",
"mod_estado": "1",
"mod_porcentaje_avance": "58.00000",
"mod_fecha_inicio": "2021-04-01 00:00:00",
"mod_fecha_fin": "2021-08-01 00:00:00",
"mod_obligatorio": null,
"mod_nota": null,
"act_cod": null,
"act_estado": null,
"estado_registro": "1",
  "obs_registro": "",
  "id_proceso_externo": "30"
}
]
}
]

```

Como requisito obligatorio para poder ejecutar el proceso de lectura de datos descrito anteriormente, dentro del sistema LMS de cada ejecutor (OTEC), deberá tener incorporado el módulo de Registro SENCE, para garantizar la seguridad e identificación de cada institución, participante y curso, con el objetivo de autenticar los datos de registro e ingreso. La información de cómo realizar esta conexión se encuentra publicada en el Manual Técnico de Integración Registro SENCE.

## Manejo de Timeouts y Rendimiento

Al interactuar con la API, es crucial considerar los timeouts para evitar interrupciones. El proceso puede llegar a ser bastante demandante tanto desde el lado del cliente, como por el lado del endpoint de SENCE, por lo que hay que evaluar setear los distintos timeouts involucrados a valores distintos a los que existen por defecto, tanto a nivel del método de envío, como el lenguaje dinámico y el motor web. Por ejemplo, si estás utilizando cURL para realizar solicitudes, puedes establecer un timeout específico como sigue:

```
curl_setopt($ch, CURLOPT_TIMEOUT, 600); // Establecer timeout en 10 minutos
```

Además, si estás utilizando PHP, puedes ajustar el tiempo de ejecución máximo mediante:

```
ini_set('max_execution_time', 600); // Establecer tiempo máximo de ejecución a 10 minutos
```

Incluimos en esta consideración también el timeout del motor web. Por ejemplo, si estás utilizando un servidor Apache, puedes configurar el timeout en el archivo de configuración "httpd.conf" o en un archivo de configuración específico del sitio virtual (virtualhost). Aquí hay un ejemplo de cómo se podría configurar el timeout en "httpd.conf":

```
Timeout 600
```

Referencias:

- [Configuración de Timeout de Apache](#)
- [cURL - CURLOPT\\_TIMEOUT](#)

## Programación Automatizada con Cron

Ya que el envío se debe realizar diariamente en horarios establecidos por el Servicio (entre 22.00 y 00.00 horas América/Santiago), se sugiere el uso de trabajos programados o Cronjobs. En el caso de Moodle, es esencial verificar que el cron nativo de Moodle se esté ejecutando sin errores. Además, para incorporar nuevas tareas, consulta la documentación de la API nativa de tareas ([https://docs.moodle.org/dev/Task\\_API](https://docs.moodle.org/dev/Task_API)).

En el caso de LMSs basados en Wordpress, puedes utilizar a su vez la función `wp_next_scheduled()`, llamando a una acción con la actividad del envío de los datos. Quedaría estructuralmente así:

```
add_action( 'sence_envio_hook', 'enviar_datos_sence' );  
wp_next_scheduled( 'sence_envio_hook' );
```

En cualquier otro contexto, podemos recurrir directamente a programar tareas automatizadas con cron en sistemas Unix, puedes agregar una nueva entrada en tu crontab usando el comando `crontab -e`. Por ejemplo, para ejecutar tu script PHP todos los días a las 22 horas, añade la siguiente línea:

```
0 22 * * * /usr/bin/php /ruta/a/tu/script.php
```

## Cálculo del Tiempo de Dedicación

Al calcular el tiempo de dedicación, se debe tener en cuenta un intervalo máximo de 45 minutos entre clics de los usuarios. Si está utilizando Moodle, le sugerimos considerar la última versión del bloque "Dedicación al Curso" [https://moodle.org/plugins/block\\_dedication/2023070300/29558](https://moodle.org/plugins/block_dedication/2023070300/29558). Esta versión cuenta con una tarea programada que calcula diariamente el tiempo de dedicación de los usuarios por curso y almacena estos valores en una tabla de la base de datos Moodle. Utilizar este bloque optimizará significativamente el rendimiento en el cálculo del tiempo de dedicación, contribuyendo a un envío más eficiente de los datos hacia el endpoint de Moodle. Esta nueva versión del bloque es compatible sólo con versiones 4 y superiores de Moodle.

Proveemos un pequeño ejemplo de cómo obtener los datos de dedicación en el caso de estar usando este bloque:

```
global $DB;

$userid = 123; // Reemplaza con el ID de usuario real

$courseid = 456; // Reemplaza con el ID de curso real

$dedicacion = $DB->get_record('block_dedication', array('userid' =>
$userid, 'courseid' => $courseid));

if ($dedicacion) {

    $tiempoDedicado = $dedicacion->timespent;

    $inicioDedicacion = $dedicacion->timestart;

} else {

    $tiempoDedicado = 0; // Establece un valor predeterminado de 0

    $inicioDedicacion = 0; // Establece un valor predeterminado de 0

}
```

## Abstracción del modelo de datos:

En el Anexo "Abstracción", se propone un modelo abstracto de las clases, funciones y loops que debería contener el programa. Se proporcionan a modo ilustrativo, con la idea de aclarar el tipo de estructura que debería tener el programa que generase este envío. En él se describe la construcción de cuatro clases principales: Curso, Alumno, Módulo y Actividad. Cada una de estas clases representa una entidad fundamental en el proceso educativo y contiene atributos que capturan información relevante sobre ellas. Por ejemplo, la clase Curso almacena detalles sobre una oferta de curso, mientras que la clase Alumno almacena información sobre un estudiante en particular. De manera similar, las clases Módulo y Actividad representan componentes esenciales del plan de estudios.

Para interactuar con la API del sistema de gestión del aprendizaje, se ha definido la clase API\_LMS. Esta clase contiene métodos que permiten enviar y recibir datos de la API. Los métodos como "enviar\_datos" se utilizan para enviar datos JSON a la API, mientras que otros métodos como "obtener\_datos\_alumno" y "obtener\_datos\_modulo" se utilizan para obtener información específica de alumnos y módulos respectivamente.

El documento proporciona un ejemplo de cómo se puede utilizar la estructura de clases para procesar datos provenientes de la API. A través de bucles anidados, se recorren los datos de cursos, alumnos, módulos y actividades para construir objetos correspondientes. Esto facilita la organización y manipulación de la información educativa en un formato estructurado.

Después de haber construido y organizado los objetos, se muestra cómo se pueden completar los atributos de cada clase con los datos obtenidos de la API. Una vez que todos los datos están estructurados adecuadamente, se puede utilizar el método "enviar\_datos" de la clase API\_LMS para enviar los datos procesados de vuelta a la API.



## Anexos

---

### Estructurar y Agrupar en Formato JSON

```
{
  "rutOtec": "75062000-3",
  "idSistema": 1350,
  "token": "0E3D8AF8-AAAA-4C15-0000-1E6E418A457E",
  "codigoOferta": "CAP-21-01-13-9999-1",
  "codigoGrupo": "CAP-21-01-13-9999-1",
  "codigoEnvio": "CAP-21-01-13-9999-1",
  "cantActividadAsincronica": 16,
  "cantActividadSincronica": 7,
  "listaAlumnos": [
    {
      "rutAlumno": 23456789,
      "dvAlumno": "5",
      "tiempoConectividad": 253115,
      "estado": 3,
      "porcentajeAvance": 95,
      "fechaInicio": "2023-07-29",
      "fechaFin": "2023-12-06",
      "fechaEjecucion": "2023-08-09",
      "evaluacionFinal": 60,
      "listaModulos": [
        {
          "codigoModulo": "ELE028837032039",
          "tiempoConectividad": 161073,
          "estado": 2,
          "porcentajeAvance": 100,
          "fechaInicio": "2023-07-29",
          "fechaFin": "2023-01-20",
          "notaModulo": 67,
          "cantActividadAsincronica": 6,
          "cantActividadSincronica": 1,
          "listaActividades": [
            {
              "codigoActividad": "Foro MC3dulo 1"
            }
          ]
        }
      ]
    }
  ]
}
```

```
},  
{  
  "codigoActividad": "Actividad Formativa 2 - Ejercicio Constituci3n de "  
},  
{  
  "codigoActividad": "Actividad Formativa 4 - GuC-a de Trabajo para la EI"  
},  
{  
  "codigoActividad": "Actividad Formativa 1 - Presentaci3n sobre Empres"  
},  
{  
  "codigoActividad": "Actividad Formativa 3 - Documentos BCIsicos que se "  
},  
{  
  "codigoActividad": "Clases SincrC3nicas -M1"  
},  
{  
  "codigoActividad": "Actividad Formativa 5 - Evaluaci3n Final MC3dulo 1"  
}  
]  
},  
{  
  "codigoModulo": "ELE028837032040",  
  "tiempoConectividad": 92042,  
  "estado": 2,  
  "porcentajeAvance": 90,  
  "fechaInicio": "2023-07-29",  
  "fechaFin": "2023-01-20",  
  "notaModulo": 53,  
  "cantActividadAsincronica": 10,  
  "cantActividadSincronica": 1,  
  "listaActividades": [  
    {  
      "codigoActividad": "Actividad Formativa 7 - TC)nicas de Atenci3n al CI"  
    },  
    {  
      "codigoActividad": "Foro MC3dulo 2"  
    },  
  ]  
}
```

```
"codigoActividad": "Actividad Formativa 6 - Presentación Normas de Ort"
},
{
  "codigoActividad": "Actividad Formativa 5 - Comparación Comunicativa"
},
{
  "codigoActividad": "Actividad Formativa 2 - Video Imagen personal, ind"
},
{
  "codigoActividad": "Actividad Formativa 1 - Imagen, Presencia y Estética"
},
{
  "codigoActividad": "Clases Sincrónicas -M2"
},
{
  "codigoActividad": "Actividad Formativa 9 - Evaluación Final Módulo 2"
},
{
  "codigoActividad": "Actividad Formativa 3 - Actividad de análisis Pres"
},
{
  "codigoActividad": "Actividad Formativa 8 - Video Expresión Escrita-Cu"
},
{
  "codigoActividad": "Actividad Formativa 4 - Presentación Técnicas de E"
}
]
}
]
},
{
  "rutAlumno": 13556884,
  "dvAlumno": "8",
  "tiempoConectividad": 134868,
  "estado": 2,
  "porcentajeAvance": 100,
  "fechaInicio": "2022-07-29",
  "fechaFin": "2023-03-06",
  "fechaEjecucion": "2023-08-09",
```

```
"evaluacionFinal": 95,  
"listaModulos": [  
  {  
    "codigoModulo": "ELE028837032039",  
    "tiempoConectividad": 85825,  
    "estado": 2,  
    "porcentajeAvance": 100,  
    "fechaInicio": "2022-07-29",  
    "fechaFin": "2023-02-09",  
    "notaModulo": 100,  
    "cantActividadAsincronica": 6,  
    "cantActividadSincronica": 1,  
    "listaActividades": [  
      {  
        "codigoActividad": "Foro MC3dulo 1"  
      },  
      {  
        "codigoActividad": "Actividad Formativa 2 - Ejercicio Constituci3n de"  
      },  
      {  
        "codigoActividad": "Actividad Formativa 4 - GuC-a de Trabajo para la EI"  
      },  
      {  
        "codigoActividad": "Actividad Formativa 1 - Presentaci3n sobre Empres"  
      },  
      {  
        "codigoActividad": "Actividad Formativa 3 - Documentos BC!sicos que se "  
      },  
      {  
        "codigoActividad": "Clases Sincr3nicas -M1"  
      },  
      {  
        "codigoActividad": "Actividad Formativa 5 - Evaluaci3n Final MC3dulo 1 "  
      }  
    ]  
  },  
  {  
    "codigoModulo": "ELE028837032040",  
    "tiempoConectividad": 49043,  
  }  
]
```

```
"estado": 2,  
"porcentajeAvance": 100,  
"fechaInicio": "2022-07-29",  
"fechaFin": "2023-02-09",  
"notaModulo": 90,  
"cantActividadAsincronica": 10,  
"cantActividadSincronica": 1,  
"listaActividades": [  
  {  
    "codigoActividad": "Actividad Formativa 7 - TC)nicas de AtenciC3n al CI"  
  },  
  {  
    "codigoActividad": "Foro MC3dulo 2"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 6 - PresentaciC3n Normas de Ort"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 5 - ComparaciC3n ComunicaciC3n a"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 2 - Video Imagen personal, ind"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 1 - Imagen, Presencia y EstC3ti"  
  },  
  {  
    "codigoActividad": "Clases SincrC3nicas -M2"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 9 - EvaluaciC3n Final MC3dulo 2"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 3 - Actividad de anC3lisis Pres"  
  },  
  {  
    "codigoActividad": "Actividad Formativa 8 - VC-deo ExpresiC3n Escrita-Cu"  
  },  
  {  
  }  
]
```

```
"codigoActividad": "Actividad Formativa 4 - Presentación Técnicas de E"  
  }  
  ]  
  }  
  ]  
  }  
  ]  
}
```

## Abstracción

```
#construimos las clases con sus atributos y métodos
class Curso:
    def __init__(self, rut_otec, id_sistema, token, codigo_oferta, codigo_grupo, codigo_envio,
                 cant_actividad_asincronica, cant_actividad_sincronica, lista_alumnos):
        self.rut_otec = rut_otec
        self.id_sistema = id_sistema
        self.token = token
        self.codigo_oferta = codigo_oferta
        self.codigo_grupo = codigo_grupo
        self.codigo_envio = codigo_envio
        self.cant_actividad_asincronica = cant_actividad_asincronica
        self.cant_actividad_sincronica = cant_actividad_sincronica
        self.lista_alumnos = lista_alumnos

class Alumno:
    def __init__(self, rut_alumno, dv_alumno, tiempo_conectividad, estado, porcentaje_avance,
                 fecha_inicio, fecha_fin, fecha_ejecucion, evaluacion_final, lista_modulos):
        self.rut_alumno = rut_alumno
        self.dv_alumno = dv_alumno
        self.tiempo_conectividad = tiempo_conectividad
        self.estado = estado
        self.porcentaje_avance = porcentaje_avance
        self.fecha_inicio = fecha_inicio
        self.fecha_fin = fecha_fin
        self.fecha_ejecucion = fecha_ejecucion
        self.evaluacion_final = evaluacion_final
        self.lista_modulos = lista_modulos
```

```
class Modulo:
    def __init__(self, codigo_modulo, tiempo_conectividad, estado, porcentaje_avance,
                 fecha_inicio, fecha_fin, nota_modulo, cant_actividad_asincronica,
                 cant_actividad_sincronica, lista_actividades):
        self.codigo_modulo = codigo_modulo
        self.tiempo_conectividad = tiempo_conectividad
        self.estado = estado
        self.porcentaje_avance = porcentaje_avance
        self.fecha_inicio = fecha_inicio
        self.fecha_fin = fecha_fin
        self.nota_modulo = nota_modulo
        self.cant_actividad_asincronica = cant_actividad_asincronica
        self.cant_actividad_sincronica = cant_actividad_sincronica
        self.lista_actividades = lista_actividades

class Actividad:
    def __init__(self, codigo_actividad):
        self.codigo_actividad = codigo_actividad

//construimos métodos para obtener datos desde el LMS
class API_LMS:
    def __init__(self, base_url, token):
        self.base_url = base_url
        self.token = token

    def enviar_datos(self, datos):
        # Código para enviar datos JSON a la API
        pass

    def obtener_datos_alumno(self, id_alumno):
        # Código para obtener datos del alumno desde la API
        pass

    def obtener_datos_modulo(self, id_modulo):
        # Código para obtener datos del módulo desde la API
        pass
```



```
def obtener_datos_actividad(self, id_actividad):
    # Código para obtener datos de la actividad desde la API
    pass
def obtener_datos_curso(self, id_curso):
    # Código para obtener datos del curso desde la API
    pass

# Ejemplo de uso
api = API_LMS("https://ejemplo.com/api", "tu_token_aqui")

# Crear objetos de alumno, módulo, actividad y curso basados en los datos JSON
# ...

# Estructura para recorrer los datos e ir generando los objetos

# Ejemplo de recorrido de la lista de cursos
for curso_data in lista_cursos:
    curso = Curso(/* ... */)

# Ejemplo de recorrido de la lista de alumnos para un curso
for alumno_data in curso_data.lista_alumnos:
    alumno = Alumno(/* ... */)

# Ejemplo de recorrido de la lista de módulos para un alumno
for modulo_data in alumno_data.lista_modulos:
    modulo = Modulo(/* ... */)

# Ejemplo de recorrido de la lista de actividades para un módulo
for actividad_data in modulo_data.lista_actividades:
    actividad = Actividad(/* ... */)

    # ... completar atributos de la clase con datos de la actividad ...

    # ... completar atributos de la clase con datos del módulo ...

    # ... completar atributos de la clase con datos del estudiante ...
```

```
# ... completar atributos de la clase con datos del curso ...  
  
# Construir la estructura de datos  
# ...  
  
# Enviar datos a la API  
# api.enviar_datos(datos)
```