

Documentación de APIs - EduFast

Índice

- [Cursos](#)
 - [Grados](#)
 - [Materias](#)
-

Cursos

GET /edufast/administrador/cursos/api/controller/

Obtiene una lista de todos los cursos disponibles en el sistema.

Método: GET

URL: /edufast/administrador/cursos/api/controller/

Descripción: Este endpoint permite recuperar una lista completa de cursos registrados en el sistema.

Request

Cuerpo de la petición: No requiere cuerpo de petición.

Response

Formato: JSON Array

Esquema de respuesta:

```
json
[
  {
    "id_cursos": "",
    "curso": "",
    "grado_id_grado": ""
  }
]
```

Parámetros de respuesta:

- `id_cursos` (string): Identificador único del curso
- `curso` (string): Nombre del curso
- `grado_id_grado` (string): Identificador del grado asociado al curso

Ejemplo de respuesta:

```
json

[
  {
    "id_cursos": "1",
    "curso": "Matemáticas",
    "grado_id_grado": "5"
  },
  {
    "id_cursos": "2",
    "curso": "Ciencias Naturales",
    "grado_id_grado": "5"
  }
]
```

POST /edufast/administrador/cursos/api/controller/

Agrega un nuevo curso al sistema.

Método: POST

URL: /edufast/administrador/cursos/api/controller/

Descripción: Este endpoint permite a los administradores agregar un nuevo curso especificando los detalles del curso, que serán almacenados en la base de datos.

Request

Cuerpo de la petición:

```
json

{
  "curso": "",
  "grado_id_grado": ""
}
```

Parámetros del cuerpo (requeridos):

- `curso` (string): Identificador del curso que se está agregando
- `grado_id_grado` (string): Identificador del grado asociado al curso

Ejemplo de petición:

```
json

{
  "curso": "1002",
  "grado_id_grado": "11"
}
```

Response

Formato: JSON Object

Esquema de respuesta:

```
json

{
  "status": "",
  "message": "",
  "data": {
    "curso": "",
    "grado_id_grado": ""
  }
}
```

Parámetros de respuesta:

- `status` (string): Indica el éxito o fallo de la petición
- `message` (string): Mensaje descriptivo sobre el resultado de la operación
- `data` (object): Contiene los detalles del curso que fue agregado, incluyendo sus identificadores

Ejemplo de respuesta exitosa:

```
json

{
  "status": "success",
  "message": "Course added successfully.",
  "data": {
    "curso": "1002",
    "grado_id_grado": "11"
  }
}
```

PUT /edufast/administrador/cursos/api/controller/

Actualiza los detalles de un curso específico en el sistema.

Método: PUT

URL: /edufast/administrador/cursos/api/controller/?action=update&id={id}

Descripción: Este endpoint permite modificar la información asociada a un curso identificado por su ID único mediante una petición PUT.

Request

Parámetros de URL:

- `action=update` (requerido): Especifica la acción de actualización
- `id` (requerido): ID único del curso a actualizar

Cuerpo de la petición:

```
json

{
  "curso": "",
  "grado_id_grado": ""
}
```

Parámetros del cuerpo:

- `curso` (string): Identificador único del curso que se desea actualizar
- `grado_id_grado` (string): ID del grado asociado al curso

Ejemplo de petición:

```
json

{
  "curso": "1002",
  "grado_id_grado": "11"
}
```

Response

Formato: JSON Object

Esquema de respuesta:

```
json
{
  "status": "",
  "message": ""
}
```

Parámetros de respuesta:

- `status` (string): Indica si la actualización fue exitosa o si hubo un error
- `message` (string): Proporciona información adicional sobre el resultado de la operación

Ejemplo de respuesta exitosa:

```
json
{
  "status": "success",
  "message": "Course updated successfully."
}
```

PATCH /edufast/administrador/cursos/api/controller/

Actualiza información específica de un curso existente.

Método: `PATCH`

URL: `/edufast/administrador/cursos/api/controller/`

Descripción: Este endpoint permite actualizar los detalles de un curso específico en el sistema mediante una petición PATCH. A diferencia del PUT, permite modificar atributos específicos sin necesidad de enviar el objeto completo del curso.

Request

Content-Type:

- `application/x-www-form-urlencoded`
- `multipart/form-data`

Parámetros del cuerpo de la petición:

- `course_id` (text, requerido): Identificador único del curso que se desea actualizar
- `title` (text): Título del curso
- `description` (text): Descripción detallada del contenido del curso

- `duration` (text): Duración del curso, típicamente en horas o días
- `price` (text): Costo de inscripción al curso

Ejemplo de petición (form-data):

```
course_id: "123"
title: "Matemáticas Avanzadas"
description: "Curso completo de matemáticas para estudiantes de bachillerato"
duration: "40 horas"
price: "$150"
```

Response

Formato: JSON Object

Esquema de respuesta:

```
json
{
  "status": "",
  "message": "",
  "updated_course": {
    "course_id": "",
    "title": "",
    "description": "",
    "duration": "",
    "price": ""
  }
}
```

Parámetros de respuesta:

- `status` (string): Indica el éxito o fallo de la operación (ej. "success" o "error")
- `message` (string): Mensaje que proporciona información adicional sobre el resultado de la petición
- `updated_course` (object): Objeto que contiene los detalles actualizados del curso, incluyendo los parámetros que fueron modificados

Ejemplo de respuesta exitosa:

json

```
{
  "status": "success",
  "message": "Course information updated successfully.",
  "updated_course": {
    "course_id": "123",
    "title": "Matemáticas Avanzadas",
    "description": "Curso completo de matemáticas para estudiantes de bachillerato",
    "duration": "40 horas",
    "price": "$150"
  }
}
```

DELETE /edufast/administrador/cursos/api/controller/

Elimina un curso específico del sistema.

Método: DELETE

URL: /edufast/administrador/cursos/api/controller/?action=delete&id={id}

Descripción: Este endpoint permite eliminar un curso específico del sistema proporcionando el ID del curso.

Request

Parámetros de URL (requeridos):

- `action=delete` (string): Especifica la acción de eliminación a realizar
- `id` (integer): Identificador único del curso que se desea eliminar

Ejemplo de petición:

http

DELETE /edufast/administrador/cursos/api/controller/?action=delete&id=15

Response

Códigos de estado HTTP:

- `200 OK`: Eliminación exitosa
- Códigos de error apropiados en caso de fallos

Estructura de respuesta: La respuesta típicamente incluye un mensaje de confirmación indicando el resultado de la operación.

Ejemplo de respuesta exitosa:

```
json
{
  "status": "success",
  "message": "Course deleted successfully."
}
```

Ejemplo de respuesta con error:

```
json
{
  "status": "error",
  "message": "Course not found or could not be deleted."
}
```

Manejo de errores:

- Si se intenta eliminar un curso inexistente, se retornará un mensaje de error apropiado
 - Otros errores potenciales incluyen restricciones de integridad referencial
-

Grados

GET /edufast/administrador/grados/api/controller/

Obtiene información relacionada con la gestión de grados del sistema.

Método: GET

URL: /edufast/administrador/grados/api/controller/

Descripción: Este endpoint permite a los usuarios interactuar con el sistema de gestión de grados para recuperar información relacionada con los grados.

Request

Parámetros: No requiere parámetros específicos en la URL o en el cuerpo de la petición.

Response

Formato: JSON Object

Descripción: La respuesta retorna un objeto JSON que contiene información relacionada con los grados. La estructura de la respuesta puede incluir varios campos que proporcionan información sobre los datos de grados gestionados por el sistema.

Uso: Este endpoint se utiliza principalmente para recuperar información relacionada con grados, que puede ser utilizada para procesamiento adicional o visualización en interfaces de usuario.

Ejemplo de respuesta:

```
json
{
  "status": "success",
  "data": [
    {
      "id_grado": "1",
      "nombre_grado": "Primero",
      "nivel": "Primaria"
    },
    {
      "id_grado": "2",
      "nombre_grado": "Segundo",
      "nivel": "Primaria"
    }
  ]
}
```

POST /edufast/administrador/grados/api/controller/

Agrega un nuevo nivel educativo y grado correspondiente al sistema.

Método: POST

URL: /edufast/administrador/grados/api/controller/

Descripción: Este endpoint permite a los usuarios agregar un nuevo nivel educativo y su grado correspondiente al sistema. Es especialmente útil para administradores que gestionan datos educativos.

Request

Cuerpo de la petición:

```
json
{
  "nivel_educativo": "",
  "grado": ""
}
```

Parámetros del cuerpo (requeridos):

- `nivel_educativo` (string): El nivel educativo que se va a agregar (ej. "Bachillerato")

- `grado` (string): El grado asociado con el nivel educativo (ej. "12°")

Ejemplo de petición:

```
json
{
  "nivel_educativo": "Bachillerato",
  "grado": "12°"
}
```

Response

Formato: JSON Object

Esquema de respuesta:

```
json
{
  "status": "",
  "message": ""
}
```

Parámetros de respuesta:

- `status` (string): Indica el éxito o fallo de la petición
- `message` (string): Mensaje descriptivo que proporciona contexto adicional sobre la operación

Ejemplo de respuesta exitosa:

```
json
{
  "status": "success",
  "message": "Educational level and grade added successfully."
}
```

Notas:

- Asegúrese de que los parámetros de entrada estén correctamente formateados según se especifica
- La API validará la entrada y responderá con mensajes apropiados para cualquier error encontrado durante el procesamiento

PUT /edufast/administrador/grados/api/controller/

Actualiza el nivel educativo y grado de un registro específico.

Método: `PUT`

URL: `/edufast/administrador/grados/api/controller/?action=update&id={id}`

Descripción: Este endpoint permite actualizar el nivel educativo y grado de un registro específico identificado por el parámetro `id`.

Request

Parámetros de URL:

- `action=update` (requerido): Especifica la acción de actualización
- `id` (requerido): Identificador único del registro a actualizar

Cuerpo de la petición:

```
json
{
  "nivel_educativo": "",
  "grado": ""
}
```

Parámetros del cuerpo:

- `nivel_educativo` (string): El nivel educativo que se establecerá (ej. "Bachillerato")
- `grado` (string): El nivel de grado que se actualizará (ej. "13°")

Ejemplo de petición:

```
json
{
  "nivel_educativo": "Bachillerato",
  "grado": "13°"
}
```

Response

Formato: JSON Object

La respuesta típicamente incluye un mensaje de éxito confirmando la actualización y los detalles del registro actualizado o una confirmación del cambio.

Ejemplo de respuesta exitosa:

json

```
{
  "status": "success",
  "message": "Educational level and grade updated successfully.",
  "data": {
    "id": "13",
    "nivel_educativo": "Bachillerato",
    "grado": "13°"
  }
}
```

Ejemplo de respuesta con error:

json

```
{
  "status": "error",
  "message": "Record not found or update failed."
}
```

Notas importantes:

- Asegúrese de que el parámetro `id` en la URL corresponda a un registro existente en la base de datos
- El nivel educativo y grado deben adherirse a los formatos esperados según se define en el esquema de la aplicación
- Si la actualización falla, la respuesta contendrá un mensaje de error detallando la razón del fallo
- Este endpoint es crucial para mantener la integridad y precisión de los registros educativos dentro del sistema

DELETE /edufast/administrador/grados/api/controller/

Elimina un registro de grado específico del sistema.

Método: DELETE

URL: /edufast/administrador/grados/api/controller/?action=delete&id={id}

Descripción: Este endpoint permite al usuario eliminar un registro de grado específico del sistema.

Request

Parámetros de URL (requeridos):

- `action=delete` (string): Especifica la acción de eliminación a realizar

- `id` (integer): Identificador único del registro de grado que se desea eliminar

Ejemplo de petición:

http

DELETE /edufast/administrador/grados/api/controller/?action=delete&id=13

Response

Códigos de estado HTTP:

- `200 OK`: Eliminación exitosa
- Códigos de error apropiados en caso de fallos

Estructura de respuesta: Tras la eliminación exitosa del registro de grado, el servidor retornará una respuesta indicando el éxito de la operación. La respuesta típicamente incluye un mensaje de estado confirmando que el grado ha sido eliminado.

Ejemplo de respuesta exitosa:

json

```
{
  "status": "success",
  "message": "Grade record deleted successfully."
}
```

Ejemplo de respuesta con error:

json

```
{
  "status": "error",
  "message": "Grade record not found or could not be deleted."
}
```

Manejo de errores:

- Si se intenta eliminar un grado inexistente, se retornará un mensaje de error apropiado
- Otros errores potenciales incluyen restricciones de integridad referencial si el grado está siendo utilizado por otros registros

Materias

GET /edufast/administrador/Materiaphp/api/controller

Obtiene información relacionada con las materias del sistema.

Método: GET

URL: /edufast/administrador/Materiaphp/api/controller

Descripción: Este endpoint se utiliza para recuperar información relacionada con el recurso "Materia" dentro de la aplicación EduFast.

Request

Parámetros: La petición GET no requiere parámetros adicionales en la URL. Sin embargo, puede soportar parámetros de consulta que pueden ser añadidos al endpoint para filtrar o personalizar la respuesta.

Response

Formato: JSON Object

Descripción: La respuesta típicamente será en formato JSON, conteniendo detalles sobre los recursos "Materia" solicitados. La estructura de la respuesta incluirá campos relevantes que describen cada entrada de "Materia", como identificadores, nombres y otros atributos asociados.

Esquema de respuesta:

```
json
{
  "materias": [
    {
      "id": "",
      "name": "",
      "description": ""
    }
  ]
}
```

Parámetros de respuesta:

- `materias` (array): Lista de materias disponibles
 - `id` (string): Identificador único de la materia
 - `name` (string): Nombre de la materia
 - `description` (string): Descripción de la materia

Ejemplo de respuesta:

```
json
{
  "materias": [
    {
      "id": "1",
      "name": "Mathematics",
      "description": "Study of numbers, shapes, and patterns."
    },
    {
      "id": "2",
      "name": "Science",
      "description": "Study of the physical and natural world."
    }
  ]
}
```

POST /edufast/administrador/Materiaphp/api/controller

Agrega una nueva materia al sistema.

Método: POST

URL: /edufast/administrador/Materiaphp/api/controller

Descripción: Este endpoint permite a los usuarios agregar una nueva materia al sistema. Requiere un payload JSON que contenga los detalles de la materia que se está agregando.

Request

Cuerpo de la petición:

```
json
{
  "materia": "",
  "area_id_area": ""
}
```

Parámetros del cuerpo (requeridos):

- materia** (string): El nombre de la materia que se va a agregar. Debe ser un título descriptivo que identifique la materia
- area_id_area** (string): El identificador del área a la cual pertenece la materia. Debe corresponder a un área existente en el sistema

Ejemplo de petición:

```
json

{
  "materia": "Algebra",
  "area_id_area": "1"
}
```

Response

Formato: JSON Object

Descripción: Tras una petición exitosa, la API retornará una respuesta indicando el estado de la operación, que típicamente incluye un mensaje de éxito y los detalles de la materia recién creada.

Ejemplo de respuesta exitosa:

```
json

{
  "status": "success",
  "message": "Subject added successfully.",
  "data": {
    "id": "3",
    "materia": "Algebra",
    "area_id_area": "1"
  }
}
```

Ejemplo de respuesta con error:

```
json

{
  "status": "error",
  "message": "Failed to add subject. Please check the provided data."
}
```

PUT /edufast/administrador/Materiaphp/api/controller/

Actualiza una materia existente en el sistema.

Método: PUT

URL: /edufast/administrador/Materiaphp/api/controller/?action=update&id={id}

Descripción: Este endpoint permite actualizar una "materia" (materia) existente en el sistema. La petición se realiza usando el método HTTP PUT y requiere el identificador de la "materia" que deseas actualizar,

especificado por el parámetro de consulta `id`.

Request

Parámetros de URL:

- `action=update` (requerido): Especifica la acción de actualización
- `id` (requerido): Identificador único de la materia a actualizar

Cuerpo de la petición:

```
json
{
  "materia": "",
  "area_id_area": ""
}
```

Parámetros del cuerpo:

- `materia` (string): El nombre de la materia que deseas actualizar
- `area_id_area` (string): El identificador del área asociada con la materia

Ejemplo de petición:

```
json
{
  "materia": "Calculo",
  "area_id_area": "1"
}
```

Response

Formato: JSON Object

Descripción: La respuesta típicamente incluirá un código de estado indicando el éxito o fallo de la operación de actualización. La estructura de la respuesta puede variar según la implementación, pero generalmente incluye información sobre el resultado de la actualización.

Parámetros de respuesta:

- `success` (boolean): Indica si la actualización fue exitosa
- `message` (string): Mensaje que proporciona información adicional sobre la operación

Ejemplo de respuesta exitosa:

json

```
{
  "success": true,
  "message": "Subject updated successfully.",
  "data": {
    "id": "2",
    "materia": "Calculo",
    "area_id_area": "1"
  }
}
```

Ejemplo de respuesta con error:

json

```
{
  "success": false,
  "message": "Failed to update subject. Subject not found or invalid data provided."
}
```

DELETE /edufast/administrador/Materiaphp/api/controller/

Elimina una materia específica del sistema.

Método: DELETE

URL: /edufast/administrador/Materiaphp/api/controller/?action=delete&id={id}

Descripción: Este endpoint se utiliza para eliminar una materia específica identificada por su ID único del sistema.

Request

Parámetros de URL (requeridos):

- action=delete (string): Especifica la acción de eliminación a realizar
- id (integer): Identificador único de la materia que se desea eliminar

Ejemplo de petición:

http

DELETE /edufast/administrador/Materiaphp/api/controller/?action=delete&id=7

Response

Códigos de estado HTTP:

- `200 OK`: Eliminación exitosa
- Códigos de error apropiados en caso de fallos

Estructura de respuesta: Tras una eliminación exitosa, la API retornará una respuesta indicando el resultado de la operación. La estructura típica de la respuesta incluye:

Parámetros de respuesta:

- `status` (string): Indica si la eliminación fue exitosa o si hubo un error
- `message` (string): Proporciona información adicional sobre el resultado de la petición de eliminación

Ejemplo de respuesta exitosa:

```
json
{
  "status": "success",
  "message": "Resource deleted successfully."
}
```

Ejemplo de respuesta con error:

```
json
{
  "status": "error",
  "message": "Resource not found or could not be deleted."
}
```

Notas importantes:

- Asegúrese de que el ID proporcionado corresponda a una materia existente; de lo contrario, la eliminación fallará
- Pueden requerirse autenticación y autorización adecuadas para realizar esta acción
- Considere las restricciones de integridad referencial si la materia está siendo utilizada por otros registros