

¿Por qué obtienes ese 400 con los tres mensajes de error?

1. slug • Definición: un “slug” es una cadena corta, legible y apta para URL que identifica de forma única un recurso (`/workspaces/mi-workspace/` en vez de `/workspaces/78c0.../`). • Regla de validación (la ves en `Workspace.slug` y en `Organization.slug`): `[a-z0-9-]+` – Solo minúsculas, números y guiones. – Ni espacios, ni mayúsculas, ni tildes, ni “_”. • Motivo del error: el JSON que enviaste contiene un carácter fuera de esa expresión (probablemente espacio, mayúscula o símbolo). Qué hacer en el front: – Normaliza el texto: `name.toLowerCase().replace(/\\s+/g, '-')` y quita acentos. – En el formulario pon un placeholder explicando qué caracteres son válidos.
2. workspace_type • Campo `workspace_type` viene con choices fijos en el modelo: `['development', 'design', 'marketing', 'sales', 'support', 'hr', 'finance', 'general']` • El backend convierte la cadena recibida en uno de esos valores; cualquier otra cosa produce “«X» is not a valid choice.”. • El valor que llega parece ser `""` (comillas escapadas) ☹ probablemente estás enviando `workspace_type: "\"development\""` en vez de `workspace_type: "development"`. Qué hacer en el front: – Usa un select cuyos options tengan exactamente esos valores. – Asegúrate de no envolverlos con comillas adicionales al serializar el JSON.
3. cover_image • El endpoint espera un fichero (campo `cover_image` del modelo es `ImageField`). • DRF solo interpreta un archivo si la petición es `multipart/form-data`. • Estás mandando el nombre o la ruta como string dentro de un JSON ☹ “The submitted data was not a file”. Qué hacer en el front: – Cambia el `Content-Type` a `multipart/form-data`. – Envía el resto de los campos como parte del mismo form-data (clave=valor) y el archivo en la clave `cover_image`.

¿Dónde entra el UUID?

- Crear Workspace POST /workspaces/ Body (form-data o JSON cuando no subas archivo):

```
{
  "organization": "", // lo pasas tú
  "name": "Plataforma Web",
  "slug": "plataforma-web",
  "workspace_type": "development",
  ...
}
```

– El organization es el UUID de la organización a la que pertenecerá este workspace (lo obtuviste del listado o del login). – El id del Workspace no lo envías; el backend crea uno nuevo (UUIDField con default=uuid4) y te lo devuelve en la respuesta.

- Crear Project POST /projects/

```
{
  "workspace": "", // la FK que enlaza con Workspace
  ...
}
```

- Más tarde, para leer/editar un Workspace usas el id que te devolvió la API: /workspaces/<WORKSPACE_UUID>/.

Resumen rápido para el equipo frontend:

1. Valida tú mismo que `slug` cumpla `[a-z0-9-]+`.
2. Envía `workspace_type` exactamente como está en la lista de choices.
3. Para `cover_image` usa `multipart/form-data`.
4. Incluye `organization` (o `workspace`) como UUID en el payload; los `id` nuevos los genera el backend.