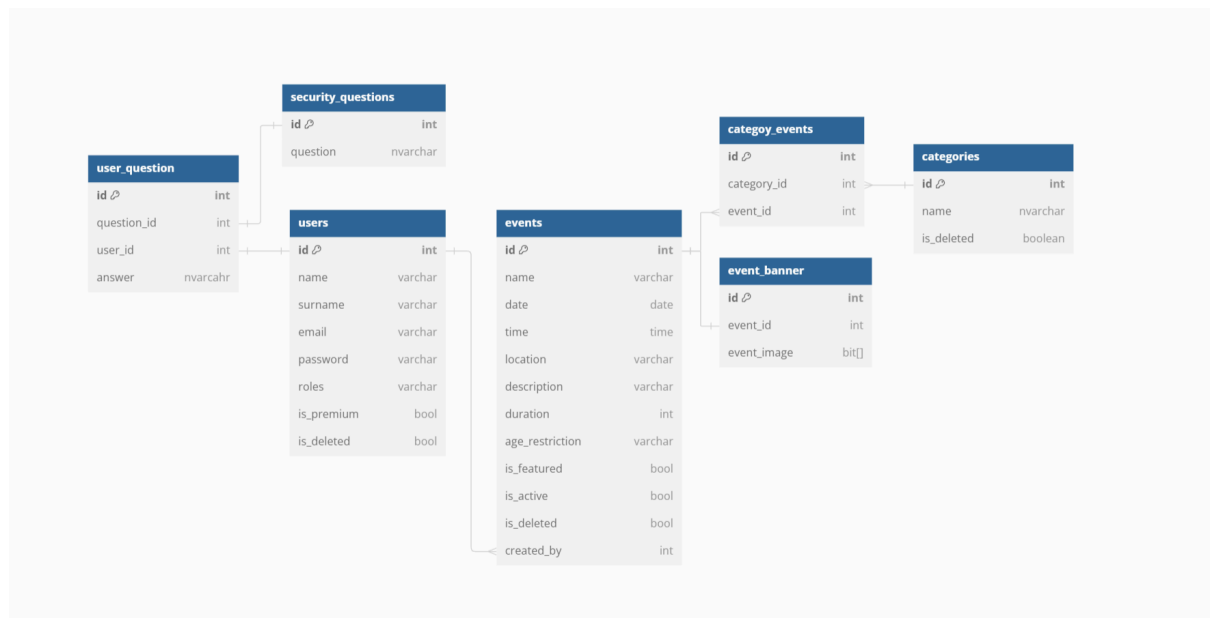


# Referencia de ProdigyPlanningAPI

## Diagrama Base de Datos



## Modulo login

### /Login (POST)

Sin necesidad de validación por token, esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- email - string
- password - string

Y devolverá, en caso de poder validar usuario y contraseña, un token JWT para acceder al resto de los endpoints.

### /Login/Signup (POST)

Sin necesidad de validación por token, esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Name - string
- Surname - string
- Email - string
- Password - string
- Roles\* - string
- UserQuestionId - int
- UserQuestionAnswer - string

*\*Este campo es opcional.*

Intentara crear un usuario con los valores otorgados devolverá un mensaje de éxito o de error. Si el campo Roles es enviado en blanco o con un valor diferente a "Organizador", al usuario creado se le asigna el rol de usuario común. El campo UserQuestionId debe

contener el id válido de una de las preguntas de seguridad almacenadas en la tabla security\_questions y UserQuestionAnswer debe ser la respuesta a la pregunta indicada anteriormente que serán utilizadas en el caso de necesitar recuperar la contraseña.

### **/Login/SignUp/GetQuestions (GET)**

Sin necesidad de validación por token, esta ruta devuelve una lista con id de las preguntas de seguridad.

## Módulo user

### **/User/GetOwnedEvents?IsActive=true (GET)**

Este recurso requiere validación de usuario a través de JWT. Esta ruta toma en formato de query de url los siguientes parámetros::

- IsActive\* - bool

*\*Este campo es opcional.*

Y devuelve una lista con los eventos creados por el usuario de sesión si el campo IsActive se deja en blanco o tiene un valor igual a FALSE. Si IsActive tiene valor TRUE se devuelve una lista con todos los eventos creados por el usuario de sesión que además se encuentren activos.

### **/User/GetRoles (GET)**

Este recurso requiere validación de usuario a través de JWT. No toma parámetros y devuelve los roles asignados al usuario de sesión.

### **/User/IsPremium (GET)**

Este recurso requiere validación de usuario a través de JWT. No toma parámetros y devuelve TRUE si el usuario es premium, o FALSE si el usuario es gratuito.

### **/User/GetUser (GET)**

Este recurso requiere validación de usuario a través de JWT. No toma parámetros y devuelve un JSON con el nombre, apellido e email del usuario de sesión.

### **/User/SetPremium (PATCH)**

Este recurso requiere validación de usuario a través de JWT. No toma parámetros. Si el usuario de sesión es premium lo convierte en gratuito, y si es gratuito lo convierte en premium.

### **/User/UpdatePassword (PATCH)**

Este recurso requiere validación de usuario a través de JWT. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- OldPassword - string
- ConfirmPassword - string
- NewPassword - string

Los valores de los parámetros OldPassword y ConfirmPassword deben ser idénticos y el parámetro de NewPassword debe ser diferente a los otros dos. Intentara modificar la contraseña del usuario de sesión y devolverá un mensaje de éxito o error.

### **/User/RecoverPassword (PATCH)**

Este recurso requiere validación de usuario a través de JWT. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Email - string
- QuestionId - int
- Answer - string

Verificará que la pregunta y respuesta de seguridad coincidan con las almacenadas en la base de datos para el usuario con el email proporcionado y de ser así generará una nueva contraseña aleatoriamente y se enviará un correo con la contraseña actualizada.

### **/User/UpdateUser (PATCH)**

Este recurso requiere validación de usuario a través de JWT. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Name\*
- Surname\*
- Email\*
- Roles\*

*\*Este campo es opcional.*

Intentará modificar los datos provistos para el usuario de sesión y devolverá un mensaje de éxito o error con los datos modificados del usuario.

### **/User/DeleteUser (DELETE)**

Este recurso requiere validación de usuario a través de JWT. Esta ruta no toma parámetros. Intentará eliminar el usuario de sesión y devolverá un mensaje de éxito o error.

## Módulo category

### **/Category (GET)**

Este recurso no requiere validación. Esta ruta no toma parámetros. Devuelve la lista de categorías existentes para eventos.

### **/Category/GetEvents?categoryId=1 (GET)**

Este recurso no requiere validación. Esta ruta toma los siguientes parámetros:

- Id - int

Intenta identificar una categoría según el id enviado y de encontrarla devolverá una lista con todos los eventos activos listados en esa categoría.

### **/Category/Add (POST)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de administrador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Name - string

Intentará crear una categoría con el nombre otorgado y devolverá un mensaje de éxito o error.

### **/Category/Edit (PATCH)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de administrador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Id - int
- NewName - string

Intentará encontrar una categoría que se llame como el valor OldName e intentará cambiar su nombre al valor NewName. Devolverá un mensaje de éxito o error.

### **/Category/Delete (DELETE)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de administrador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- id - int

Intentará identificar una categoría según el id proporcionado y luego intentará eliminarla. Devolverá un mensaje de éxito o error.

## Módulo event

### **/Event (GET)**

Este recurso no requiere validación. Esta ruta no toma parámetros. Devuelve la lista de eventos existentes.

### **/Event/GetById?id=1 (GET)**

Este recurso no requiere validación. Esta ruta toma los siguientes parámetros:

- Id - int

Intentará identificar un evento según el id proporcionado. Devuelve una respuesta con los datos del evento.

### **/Event/GetByPeriod?cdPeriod=072024 (GET)**

Este recurso no requiere validación. Esta ruta los siguientes parámetros:

- Cd\* - int (formato de periodo: mmyyyy)

*\*Este campo es opcional.*

Si el periodo es enviado en el formato correcto devolverá una lista de eventos que ocurran en el periodo indicado. Si no se envía ningún código de periodo se devuelve una lista de eventos correspondientes al periodo actual.

### **/Event/GetByFilters?name=Evento1&category=1&fromDate=072024&toDate=082024&isActive=true (GET)**

Este recurso no requiere validación. Esta ruta toma los siguientes parámetros:

- name\* - string
- categoryId\* - int
- fromDate\* - int (formato de fecha: dd/mm/yyyy)
- toDate\* - int (formato de fecha: dd/mm/yyyy)
- isActive\* - bool

*\*Este campo es opcional.*

Devuelve una lista de eventos según los filtros proporcionados. Todos los campos son opcionales pero debe haber por lo menos uno. El campo Name debe ser una coincidencia

exacta con el nombre del evento para encontrarlo. Si el campo IsActive contiene un valor TRUE devolverá una lista de todos los eventos que coincidan con los demás filtros y que además se encuentren activos, de lo contrario se devolverá una lista de todos los eventos incluso los finalizados.

### **/Event/GetFeatured (GET)**

Este recurso no requiere validación. Esta ruta no toma parámetros. Devuelve la lista de eventos existentes que se encuentren destacados.

### **/Event/Add (POST)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Name - string
- Location - string
- Date - string (formato de fecha: dd/mm/yyyy)
- Time - string (formato de hora: hh:mm:ss)
- Duration - int
- Description\* - string
- CategoryId - int

*\*Este campo es opcional.*

Intentara crear un evento con los valores otorgados y devolverá un mensaje de éxito o error.

### **/Event/Edit (PATCH)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Id - int
- NewName\* - string
- NewDescription\* - string
- NewDate\* - string (formato de fecha: dd/mm/yyyy)
- NewTime\* - string (formato de hora hh:mm:ss)
- NewLocation\* - string
- NewCategoryId\* - int
- NewDuration\* - int

*\*Este campo es opcional.*

Solo el usuario creador del evento puede modificarlo. Intentara modificar los datos provistos para el evento que cuyo id coincida con el valor id y devolverá un mensaje de éxito o error. La categoría agregada en el campo NewCategoryId no reemplaza a la actual sino que se agrega a la lista.

### **/Event/SetFeatured (Patch)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- id - int

Solo el usuario creador del evento puede cambiar si un evento está destacado o no. si el evento es un evento destacado, le quita el destacado; si no es destacado, lo destaca..

### **/Event/Delete (DELETE)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- id - int

Solo el usuario creador del evento puede eliminarlo. Intentara eliminar el evento que cuyo nombre coincida con el valor Name y devolverá un mensaje de éxito o error.

### **/Event/GetBanner?id=1 (GET)**

Este recurso no requiere validación. Esta ruta toma los siguientes parámetros:

- Id - int

Intentara buscar un evento cuyo id coincida con el enviado en la petición, si lo encuentra y este tiene un banner se devolverá una respuesta con el banner en formato imagen base64.

### **/Event/AddBanner (POST)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición Form-Data los siguientes parámetros:

- Id - int
- FormFile - imagen en formato .png con una dimensión máxima de 728x90px

Se intentará encontrar un evento cuyo id coincida con el enviado en la petición, de encontrarlo se verificará que el creador del evento sea el usuario de sesión y de ser así se cargara al evento la imagen proporcionada, y se devolverá un mensaje de éxito o error.

### **/Event/RemoveBanner (DELETE)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Id - int

Intentara buscar un evento cuyo id coincida con el enviado en la petición, si lo encuentra y este tiene un banner se eliminará y se devolverá una respuesta con mensaje de éxito.

### **/Event/AddCategory (PATCH)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Id - int
- CategoryId - int

Se intentará encontrar un evento cuyo id coincida con el enviado en la petición, se valida que el usuario de sesión sea el creador del evento. Se intentará encontrar una categoría cuyo id coincida con el enviado en la petición, de encontrarse se agregara esa categoría al evento y se devolverá un mensaje de éxito. Un evento puede contar con un máximo de 2 categorías.

### **/Event/RemoveCategory (PATCH)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- Id - int
- CategoryId - int

Se intentará encontrar un evento cuyo id coincida con el enviado en la petición, se valida que el usuario de sesión sea el creador del evento. Se intentará encontrar una categoría cuyo id coincida con el enviado en la petición, de encontrarse se verifica que esa categoría forme parte de las categorías del evento, si esto es así, se quitara esa categoría al evento y se devolverá un mensaje de éxito. Un evento no puede contar con menos de 1 categoría.

### **/Event/ReplaceCategory (PATCH)**

Este recurso requiere validación de usuario a través de JWT y puede ser accedido por usuarios con rol de organizador. Esta ruta toma en el cuerpo de la petición JSON los siguientes parámetros:

- EventId - int
- ReplaceId - int
- ReplacementId - int

Se intentará encontrar un evento cuyo id coincida con el enviado en la petición, se valida que el usuario de sesión sea el creador del evento. Se intentará encontrar una categoría cuyo id coincida con el parámetro ReplaceId enviado en la petición, de encontrarse se verifica que esa categoría forme parte de las categorías del evento. Se intentará encontrar una categoría cuyo id coincida con el parámetro ReplacementId y se verifica que esta categoría no forme parte de las categorías del evento. De darse todas las validaciones se quita de la lista de categorías del evento la categoría cuyo id coincide con ReplaceId y se agrega aquella cuyo id coincide con ReplacementId