

GastroGuide

Gastro Guide es la nueva aplicación de moda para encontrar tus restaurantes favoritos.

Version 1.0.0

Paths

/restaurantes

GET /restaurantes

Summary

Get Restaurantes

Description

El endpoint devuelve un array de objetos Restaurante que encajan en los criterios de búsqueda.

Parameters

Name	Located in	Description	Required	Schema
tipoCocinalID	query	Query que filtra restaurantes por tipo de cocina	No	→number
tipoAmbientelID	query	Query que filtra restaurantes por tipo de ambiente	No	→number
localidad	query	Query que filtra restaurantes por localidad	No	→string
valoracion	query	Query que filtra restaurantes por valoración	No	→string

Responses

Code	Description	Schema
200	Array de Restaurantes	$\Rightarrow [$ $\quad \rightarrow \text{Restaurante} \{ \}$ $]$
default	Unexpected error	\Rightarrow $\quad \rightarrow \text{Error} \{$ $\quad \quad \text{codigo: } \rightarrow \text{integer}$ $\quad \quad \text{mensaje: string}$ $\quad \quad \text{campos: string}$ $\quad \}$

Try this operation

POST /restaurantes

Summary

Post Restaurantes

Description

El endpoint permite añadir un restaurante nuevo a la colección.

Parameters

Name	Located in	Description	Required Schema
Restaurante	body	El objeto que contiene todos los datos del restaurante.	<p>Yes ↳</p> <pre> ▼ Restaurante { restaurante: string _id: number nombre: string apellidos: string genero: string email: string password: string telefono: number localidad: string valoracion: number tipoCocinaID: ▶ [] tipoAmbienteID: ▶ [] adresa: string tipo: ▶ [] imagen: string menu: ▶ [] opiniones: ▶ [] promociones: ▶ Promociones { } recetas: ▶ Recetas { } } </pre>

Responses

Code	Description	Schema
200	Mensaje de respuesta	<pre> ▼ Mensaje { mensaje: string } </pre>
default	Unexpected error	<pre> ▼ Error { codigo: ▶ integer mensaje: string campos: string } </pre>

Try this operation

/restaurantes/{restauranteId}

GET /restaurantes/{restauranteId}

Summary

Get Restaurante especifico

Description

El endpoint devuelve un Restaurante especifico.

Parameters

Name	Located in	Required	Schema
restauranteId	path	Yes	↔ string

Responses

Code	Description	Schema
200	Restaurante	<pre> ▼ Restaurante { restaurante: string _id: number nombre: string apellidos: string genero: string email: string password: string telefono: number localidad: string valoracion: number tipoCocinaID: ▶ [] tipoAmbienteID: ▶ [] adresa: string tipo: ▶ [] imagen: string menu: ▶ [] opiniones: ▶ [] promociones: ▶ Promociones { } recetas: ▶ Recetas { } } </pre>
default	Unexpected error	<pre> ▼ Error { codigo: ▶ integer mensaje: string campos: string } </pre>

Try this operation

POST /restaurantes/{restauranteId}

Summary

Post Restaurante especifico

Description

El endpoint permite modificar un Restaurante especifico.

Parameters

Name	Located in	Description	Required	Schema
restauranteId	path		Yes	☞ string ▼ Restaurante { restaurante: string _id: number nombre: string apellidos: string genero: string email: string password: string telefono: number localidad: string valoracion: number tipoCocinaID: ▶ [] tipoAmbienteID: ▶ [] adresa: string tipo: ▶ [] imagen: string menu: ▶ [] opiniones: ▶ [] promociones: ▶ Promociones { } recetas: ▶ Recetas { } }
Restaurante	body	El objeto que contiene todos los datos del restaurante.	No	☞ ▼ Restaurante { restaurante: string _id: number nombre: string apellidos: string genero: string email: string password: string telefono: number localidad: string valoracion: number tipoCocinaID: ▶ [] tipoAmbienteID: ▶ [] adresa: string tipo: ▶ [] imagen: string menu: ▶ [] opiniones: ▶ [] promociones: ▶ Promociones { } recetas: ▶ Recetas { } }

Responses

Code	Description	Schema
200	Mensaje de respuesta	☞ ▼ Mensaje { mensaje: string }
default	Unexpected error	☞ ▼ Error { codigo: ▶ integer mensaje: string campos: string }

Try this operation

/login

GET /login

Summary

LogIn

Description

El endpoint devuelve un objeto Restaurante dependiendo del correo y la contraseña enviados.

Parameters

Name	Located in	Description	Required Schema
Credentials	body	Correo electrónico y contraseña del restaurante	Yes ↔ <pre>▼ Login { email: string password: string }</pre>

Responses

Code	Description	Schema
200	Restaurante	↔ <pre>▼ Restaurante { restaurante: string _id: number nombre: string apellidos: string genero: string email: string password: string telefono: number localidad: string valoracion: number tipoCocinaID: ▶ [] tipoAmbienteID: ▶ [] адреса: string тип: ▶ [] изображение: string меню: ▶ [] отзывы: ▶ [] промоции: ▶ Promociones { } рецепты: ▶ Recetas { } }</pre>
default	Unexpected error	↔ <pre>▼ Error { código: ▶ integer mensaje: string campos: string }</pre>

Try this operation

Models

Restaurante

```
« ▼Restaurante {  
    restaurante: string  
    _id: number  
    nombre: string  
    apellidos: string  
    genero: string  
    email: string  
    password: string  
    telefono: number  
    localidad: string  
    valoracion: number  
    tipoCocinaID: ▼ [  
        number  
    ]  
    tipoAmbienteID: ▼ [  
        number  
    ]  
    adresa: string  
    tipo: ▼ [  
        string  
    ]  
    imagen: string  
    menu: ▼ [  
        ▼Menu {  
            titol: string  
            platos: string  
        }  
    ]  
    opiniones: ▼ [  
        ▼Opiniones {  
            imagen: string  
            nombre: string  
            comentario: string  
            valoracion: number  
        }  
    ]  
    promociones: ▼Promociones {  
        promocion: string  
        descripcion: string  
    }  
    recetas: ▼Recetas {  
        imagen: string  
        nombre: string  
        preparacion: string
```

```
        ingredientes: ▶ [
            string
        ]
    }
}
```

Promociones

```
▼ Promociones {
    promocion: string
    ↳ descripcion: string
}
```

Recetas

```
▼ Recetas {
    imagen: string
    nombre: string
    ↳ preparacion: string
    ↳ ingredientes: ▶ [
        string
    ]
}
```

Opiniones

```
▼ Opiniones {
    imagen: string
    nombre: string
    ↳ comentario: string
    valoracion: number
}
```

Menu

```
▼ Menu {
    titol: string
    ↳ platos: string
}
```

Error

```
▼ Error {
    codigo: ▶ integer
    ↳ mensaje: string
    campos: string
}
```

Mensaje

```
▼ Mensaje {
    ↳ mensaje: string
}
```

Login

```
▼ Login {  
  ↵   email:   string  
  ↵   password: string  
}  
}
```