

## EJEMPLOS JSON

### Crear job

**POST api/jobs**

```
{
  "Name": "Bienvenido al reto angular",
  "Info": "Siga la guía proporcionada",
  "Lat": 8.975129537993919,
  "Lng": -79.52674082480371,
  "ContactID": 123,
}
```

**\*\*Si la llamada es satisfactoria recibirá como respuesta el objeto creado, de lo contrario recibirá null.**  
Pasar un ContactID que no exista es una causa común de error.

### Editar job

**PUT api/jobs**

```
{
  "Name": "Bienvenido al reto angular",
  "Info": "Siga la guía proporcionada",
  "Lat": 8.975129537993919,
  "Lng": -79.52674082480371,
  "JobID": 161,
  "ts": "AAAAAAAOq0="
}
```

**\*\***Cuando se está actualizando un objeto es importante enviar el JobID, que debe existir, y el ts. El ts es una propiedad que identifica la última vez que se modificó el objeto. El backend solo permitirá la actualización si se envía el último ts conocido por la base de datos, haciendo así el manejo de concurrencias. El ts se obtiene en las respuestas del GET, POST y PUT.

### Crear contact

#### **POST api/contacts**

```
{
  "ContactTypeID": 1,
  "ContactInfo":{
    "FirstName":"string",
    "LastName":"string",
    "Email":"string",
    "Phone1":"string",
    "Phone2":"string",
    "Title":"string",
    "Info":"string",
    "Company":"string"
  }
}
```

**\*\***Use siempre ContactTypeID=1 para este ejercicio.

## Editar contact

```
{
  "ContactID": 12,
  "ContactTypeID": 1,
  "ContactInfo": {
    "ContactInfoID": 28,
    "ts": "AAAAAAAO3O=",
    "FirstName": "string",
    "LastName": "string",
    "Email": "string",
    "Phone1": "string",
    "Phone2": "string",
    "Title": "string",
    "Info": "string",
    "Company": "string"
  }
}
```

**\*\*ContactInfoID** solo necesita estar declarada una vez y dentro de la propiedad anidada ContactInfo. El campo ts no es necesario afuera de ContactInfo.