

# Clase 15: Handlebars

Sitio: [Centro de E-Learning - UTN.BA](#)  
Curso: Curso de Backend Developer - Turno  
Noche  
Libro: Clase 15: Handlebars

Imprimido  
por:  
Día:

Nilo Crespi

Friday, 23 de January de 2026,  
10:29

## Tabla de contenidos

1. Introducción
2. Handlebars
3. ¡A practicar!

# 1. Introducción

Handlebars es un sistema de plantillas que facilita la separación entre la lógica del código y la presentación en aplicaciones web. Su integración con Express permite generar contenido dinámico en HTML a partir de datos en formato JSON, lo que hace que el desarrollo sea más organizado y eficiente. Gracias a su sintaxis simple y la posibilidad de reutilizar fragmentos de código mediante "partials", se convierte en una herramienta útil para gestionar la vista de una aplicación sin mezclarla con la lógica del servidor.

Al utilizar Handlebars con Express, es posible renderizar vistas dinámicas y pasar datos desde el servidor al cliente de manera sencilla. Además, permite definir plantillas reutilizables, como encabezados y pies de página, optimizando la estructura del código. Esto lo convierte en una opción popular para desarrollar aplicaciones web escalables y mantenibles.

## 2. Handlebars

### Handlebars

- Es un sistema de plantillas Javascript basado en Mustache Templates.
- Mantiene separados el código y la vista.
- Permite generar HTML a partir de objetos con datos en formato JSON.



### Instalación

Adaptador para express: <https://github.com/pillarjs/hbs>

```
$ npm install hbs
```

### Uso

El uso de hbs como motor de visualización predeterminado requiere solo una línea de código en la configuración de la aplicación. Esto generará archivos .hbs cuando se llame a `res.render`.

```
const hbs = require('hbs');
```

### `res.render()`

La función `res.render()` se usa para renderizar una vista y envía la cadena HTML renderizada al cliente.

```
app.get('/', (req, res) => {  
  res.render('home');  
})
```

### Argumentos desde el controlador

### En el controlador:

```
app.get('/', (req, res)=>{  
  res.render('home', {  
    nombre: 'Cosme Fulanito',  
    titulo: 'Travel Coffe'  
  });  
})
```

### En la vista:

```
<title>{{titulo}}</title>
```

## Partials

Crear parciales nos permite reutilizar código.

```
//Handlebars  
app.set('view engine', 'hbs');  
hbs.registerPartials( __dirname + '/views/partials');
```

### En las vistas:

```
<!-- Header -->  
{{> header }}
```

### 3. ¡A practicar!

#### ¡A practicar!

1. Establecé dos páginas con sus respectivas vistas utilizando `res.render()` y con los mismos parámetros.
2. Creá un parcial llamado header y footer.
3. Luego, implementá los parciales en las dos páginas. Recordá utilizar `{{ }}`
4. Creá un formulario para agregar nuevos usuarios y mostrar la lista de usuarios en una vista. Deben utilizar rutas, `partials` y `helpers`.