



# Unidad # 3

## Incorporando tecnologías de Servidor Web

Programación Computacional IV

Gisela Yasmín García Espinoza



**UNIVERSIDAD  
GERARDO BARRIOS**  
Líderes en Gestión del Conocimiento





## Competencia de la asignatura

Aplicar los lenguajes de programación y tecnologías para la Web con el propósito de utilizarlos en el desarrollo de aplicaciones basándose en estándares internacionales de desarrollo de software trabajando de forma individual y colaborativa.

# Contenidos a desarrollar

- Modulos en Node.js
- Introducción a Express



# Módulos en Node.js



# Módulos en Node.js

Los módulos en el node.js un conjunto de funciones y objetos de JavaScript que las aplicaciones externas pueden usar. La descripción de un fragmento de código como módulo se refiere menos a lo que es el código que a lo que hace; cualquier archivo de Node.js o colección de archivos puede considerarse un módulo si sus funciones y datos se pueden utilizar en programas externos.

# Crear un módulo en Node.js

El módulo a crear contendrá los nombres de los departamentos de El Salvador.

Para el ejemplo se creara una clase que permitirá crear objetos a los cuales se asignara un nombre y un alias; adicional se obtendrá un departamento al azar y se exportara.

# Crear un módulo en Node.js

```
class Department{  
  constructor(name, alias){  
    this.name = name;  
    this.alias = alias;  
  }  
}
```

Clase base para crear departamentos

```
exports.randomDepartment = () => {  
  return allDepartments[Math.floor(Math.random() * allDepartments.length)]  
}  
  
exports.allDepartments = allDepartments;
```

Exportando módulo

```
//agregando instancias para crear los elementos  
const allDepartments = [  
  new Department('San Miguel', 'SM'),  
  new Department('San Salvador', 'SS'),  
  new Department('Usulután', 'US'),  
  new Department('Santa Ana', 'SA')  
];
```

Crear departamentos haciendo uso de clase Department

# Crear un módulo en Node.js

Una vez creado el módulo se procede a utilizarlo.

Para confirmar que su módulo funciona. En este paso, usará el REPL para cargar el módulo colors.

Mientras se encuentra en el REPL

```
PS C:\Users\Gisela Espinoza\Documents\modulesExample> node
Welcome to Node.js v16.14.2.
Type ".help" for more information.
> departments = require('./departments');
{
  randomDepartment: [Function (anonymous)],
  allDepartments: [
    Department { name: 'San Miguel', alias: 'SM' },
    Department { name: 'San Salvador', alias: 'SS' },
    Department { name: 'Usulután', alias: 'US' },
    Department { name: 'Santa Ana', alias: 'SA' }
  ]
}
> 
```



# Crear un módulo en Node.js

Lo siguiente es guardar el modulo como dependencia.

1. Cree una carpeta fuera de la carpeta de Proyecto
2. Iniciar la carpeta con npm, se generara el archivo package.json

```
PS C:\Users\Gisela Espinoza\Documents\appDepartments> mkdir appDepartments  
PS C:\Users\Gisela Espinoza\Documents\appDepartments> |
```

```
PS C:\Users\Gisela Espinoza\Documents\appDepartments> npm init -y  
Wrote to C:\Users\Gisela Espinoza\Documents\appDepartments\package.json:  
  
{  
  "name": "appdepartments",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC"  
}  
  
PS C:\Users\Gisela Espinoza\Documents\appDepartments> |
```

# Crear un módulo en Node.js

2. Instalar el modulo departments dentro de la carpeta modulesExample, utilizando el indicador -save

Guardar modulo local como dependencia

```
PS C:\Users\Gisela Espinoza\Documents\appDepartments> npm install --save ../modulesExample

added 1 package, and audited 3 packages in 1s

found 0 vulnerabilities
PS C:\Users\Gisela Espinoza\Documents\appDepartments> 
```

```
package.json X
appDepartments > {} package.json > ...
1  {
2    "name": "appdepartments",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "modulesexample": "file:../modulesExample"
14   }
15 }
16
```

# Crear un módulo en Node.js

## 3. Listando módulos instalados

```
Found 0 vulnerabilities
PS C:\Users\Gisela Espinoza\Documents\appDepartments> ls node_modules

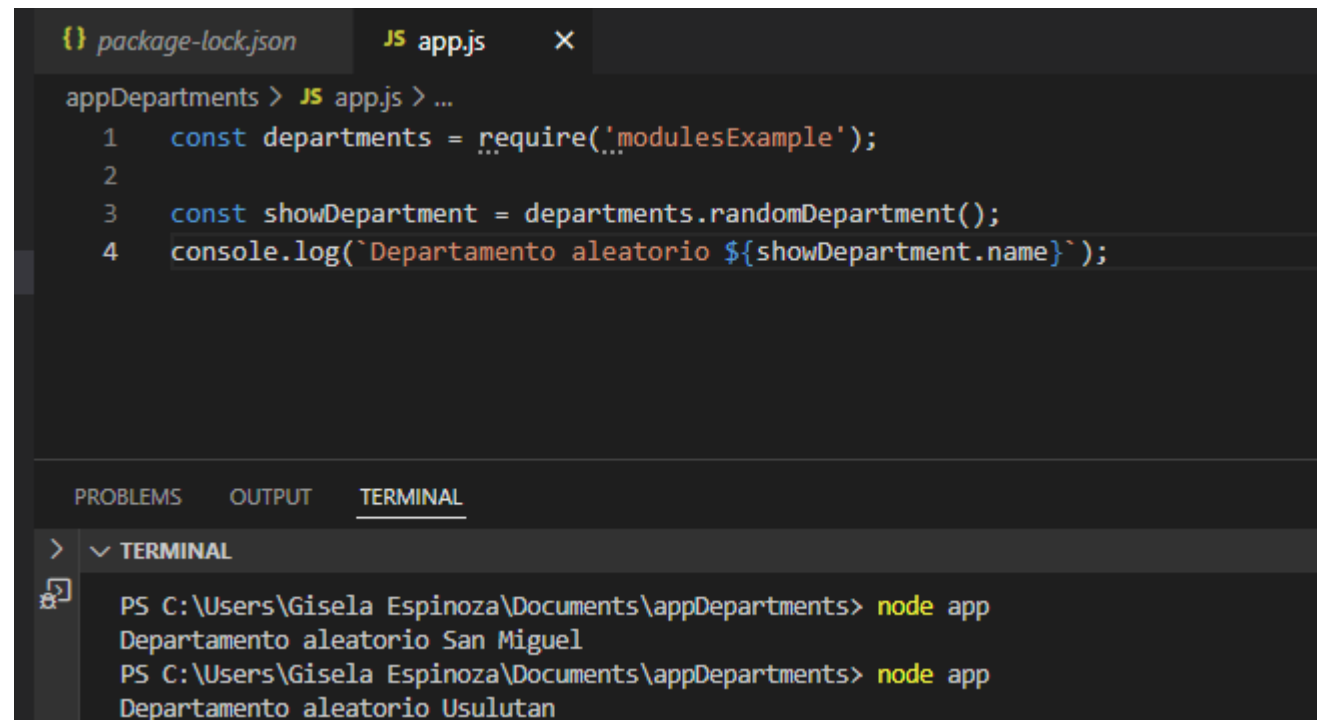
Directory: C:\Users\Gisela Espinoza\Documents\appDepartments\node_modules

Mode                LastWriteTime         Length Name
----                -
d----l            4/30/2022   8:28 AM             modulesexample
-a----            4/30/2022   8:28 AM             302 .package-lock.json

PS C:\Users\Gisela Espinoza\Documents\appDepartments> |
```

# Crear un módulo en Node.js

## 4. Utilizando modulo local instalado



The screenshot shows a VS Code editor with a file named `app.js` open. The code in the file is as follows:

```
1  const departments = require('modulesExample');
2
3  const showDepartment = departments.randomDepartment();
4  console.log(`Departamento aleatorio ${showDepartment.name}`);
```

Below the editor, the TERMINAL panel is active, showing the execution of the script:

```
PS C:\Users\Gisela Espinoza\Documents\appDepartments> node app
Departamento aleatorio San Miguel
PS C:\Users\Gisela Espinoza\Documents\appDepartments> node app
Departamento aleatorio Usulután
```

# Módulos en Node.js

- **Express framework** – Express es un framework de aplicaciones web Node js mínimo y flexible que proporciona un sólido conjunto de funciones para la web y las aplicaciones móviles.
- **Socket.io – Socket.IO** permite la comunicación bidireccional basada en eventos en tiempo real. Este módulo es bueno para la creación de aplicaciones basadas en chat.
- **Jade -Jade** es un motor de plantillas de alto rendimiento e implementado con JavaScript para nodes y navegadores.
- **MongoDB** – El controlador MongoDB Node.js es el controlador node.js oficialmente compatible para MongoDB.
- **Restify** – restify es un framework liviano, similar a express para crear API REST
- **Bluebird** – Bluebird es una biblioteca prometedora con todas las funciones y se centra en funciones y rendimiento innovadores

# Express

Es el framework web más popular de Node, un marco de aplicación web de node.js mínimo y flexible, que proporciona un conjunto sólido de funciones para crear aplicaciones web de una sola página, de varias páginas e híbridas

- Escritura de manejadores de peticiones con diferentes verbos HTTP en diferentes caminos URL (rutas).
- Integración con motores de renderización de "vistas" para generar respuestas mediante la introducción de datos en plantillas.
- Establecer ajustes de aplicaciones web como qué puerto usar para conectar, y la localización de las plantillas que se utilizan para renderizar la respuesta.
- Añadir procesamiento de peticiones "middleware" adicional en cualquier punto dentro de la tubería de manejo de la petición.

# Uso de Express

La siguiente línea de comando muestra cómo se puede instalar un módulo "express".

```
C:\Users\Gisela>npm install express
npm notice
npm notice New minor version of npm available! 7.7.6 -> 7.11.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v7.11.1
npm notice Run npm install -g npm@7.11.1 to update!
npm notice

added 50 packages, and audited 51 packages in 10s

found 0 vulnerabilities
```

El comando anterior descargará los archivos necesarios que contienen los "módulos express" y también se encargará de la instalación.

Una vez que se ha instalado el módulo, para usar un módulo en una aplicación Node.js, se utiliza la palabra clave 'require'. Esta palabra clave es una forma que utiliza Node.js para incorporar la funcionalidad de un módulo en una aplicación.

# Uso de Express

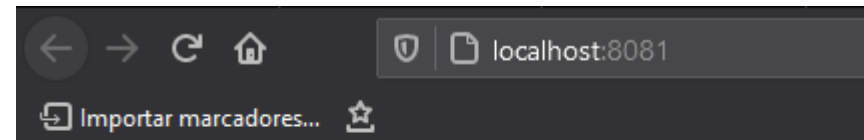
## Ejemplo

JS ejemplo.js

C: > Users > Gisela > Documents > JS ejemplo.js > ...

```
1 var express = require('express'); //se incluye modulo express
2 var app = express(); //se crea un objeto de express
3
4
5 app.get('/', function(req, res) { //ruta que se llamará cuando se reciba una petición HTTP GET con una dirección ('/')
6   res.send('Express - Programacion Computacional IV!');
7 });
8
9 app.listen(8081, function() { //define y crea el servidor, escuchando el puerto 8081
10   console.log('Ejecutando en puerto 8081!');
11 });
```

```
PS C:\Users\Gisela\Documents> node ejemplo.js
Ejecutando en puerto 8081!
```



Express - Programacion Computacional IV!



# Creando módulos

Node.js tiene la capacidad de crear módulos personalizados permitiendo incluir esos módulos personalizados en una aplicación Node.js.

## Ejemplo:

- Crear un archivo llamado modulo.js

```
JS modulo.js X
JS modulo.js > ...
1  var exports=module.exports={}; //garantiza que los demas archivos puedan acceder a la
2  //funcionalidad de este archivo
3
4  exports.suma=function(a,b) //se define una funcion llamada suma con dos parametros
5  {
6  |   return a+b; //devuelve la suma de los parametros
7  };
```

# Creando módulos

Crear un archivo llamado principal.js

```
JS principal.js ●  
JS principal.js > ...  
1  var modulo=require('./modulo.js'); //se incluye la funcionalidad de modulo.js  
2  console.log("El resultado de la suma es: " + modulo.suma(6,8)); //llamado al afuncion suma
```

Resultado

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  
  
PS C:\Users\Gisela\Documents\node> node principal.js  
El resultado de la suma es: 14
```

## Tabla de actividades

Nombre de la actividad	Módulos
Tipo de actividad	Investigación
Tipo de participación	Individual
Competencias específica de la asignatura	Integrar tecnologías y arquitecturas web, paradigmas de programación y gestión de datos con DBMS para desarrollar aplicaciones web mediante el lenguaje de programación PHP trabajando con un enfoque lógico y aplicando las buenas prácticas de programación de manera individual y colaborativa.
Instrucciones	Después de dar lectura al contenido de la semana, y realizar ejercicios prácticos ejemplo, deberá: Desarrollar una aplicación que permita leer archivos y guardar información haciendo uso de los módulos nativos de Node.js
Fecha de entrega	La fecha límite de participación será el domingo al final de la semana, a las 11:59 p.m.
Instrumento de evaluación	Lista de cotejo
Ponderación	50% Laboratorio I

## Recursos Complementarios

Recurso	Título	Cita Referencial
Sitio Web	Express Infraestructura web rápida, minimalista y flexible para Node.js	<a href="https://expressjs.com/es/">https://expressjs.com/es/</a>
Sitio Web	Cómo crear un módulo Node.js	<a href="https://www.digitalocean.com/community/tutorials/how-to-create-a-node-js-module-es">https://www.digitalocean.com/community/tutorials/how-to-create-a-node-js-module-es</a>

¿Preguntas?

# ¡Muchas gracias!