Temas vistos

[**Clase # 21 - Base de datos - 20/5**](#_oas2i4rhpt1y)

[Temas](#_jingarnq0u80)

[Práctica](#_b0c6of66l18)

[Material Didáctico](#_z4n6iopuh9jp)

[**Clase # 22 - Base de datos parte 2 | 22/5**](#_ia41ikkwai9)

[Temas](#_jmyjnjtabh6l)

[Práctica](#_6eg918fbkeqf)

[Material didáctico](#_2ko5f8h8ddfe)

[**Clase # 23 - Base de datos parte 3 | 27/5**](#_n17g1rltnqsx)

[Temas](#_c5z84yvis1qb)

[Material didáctico](#_qrb5rwt030v5)

[**Clase # 24 - Introducción a NodeJS 1 - 29/5**](#_5c0w5y3ojzfw)

[Temas](#_6khe2zhg9v4y)

[Práctica](#_l4bjmpchx4sq)

[Material didáctico](#_w8qb3h82m82k)

[**Clase # 25 - NodeJS 2 - Módulos | 3/6**](#_alfm6m3uj5s4)

[Temas](#_kl994tqdwztz)

[Material didáctico](#_idm4brtkjyv6)

[**Clase # 26 - NodeJS 3 - Rutas | 5/6**](#_3hecmr74orlk)

[Temas](#_39or59gufxk7)

[Práctica](#_jta1qi1fvl8e)

[Material didáctico](#_98mfdo7ewggh)

[**Clase # 27 - NodeJS 4 - Rutas parte 2 | 10/6**](#_ri45p6tbf35)

[Temas](#_4ybidq9bani3)

[Material didáctico](#_saptbqrf1iut)

[**Clase # 28 - NodeJS 5 | 12/6**](#_wvfjuahod0zx)

[Temas](#_r3ka40yk4igu)

[Práctica](#_ivy12s1e2m4r)

[Material didáctico](#_q00xaxh8v3zb)

[**Clase # 29 - NodeJS 6 | 19/6**](#_i5s26dhib2v0)

[Temas](#_dxjufc6xcuhf)

[Material didáctico](#_djuw4r7x4jyp)

[**Clase # 30 - NodeJS 7 | 24/6**](#_4fy6pnnawsal)

[Temas](#_u063okil4szr)

[Material didáctico](#_5ux2fl1rnkuv)

[**Clase # 31 - NodeJS 8 | 26/6**](#_tq0zj8i76jyz)

[Temas](#_rsbb18vrccmn)

[Material didáctico](#_lvxl7sen5bas)

[**Clase # 32 - Presentación de trabajos #1 | 1/7**](#_of0mfks5big6)

[Temas](#_tw85bf416oc3)

[**Clase # 33 - Presentación de trabajos #2 | 3/7**](#_hf8bww3blwi5)

[Temas](#_v5hx0jq6n6at)

[**Clase # 34 - Presentación de trabajos #3 | 8/7**](#_r4eouigbo2oc)

[Temas](#_x7618we9se7i)

[**Clase # 35 - Cierre | 10/7**](#_ry4him1t1w2d)

[Temas](#_nmlredk6w9r)

[Material](#_8fyoc44kav54)

# Clase # 21 - Base de datos - 20/5

### Temas

* Qué es una base de datos
* Tipos de base de datos (relacional y no relacional)
* roadmaps
* Instalación de Xampp
* Lógica de una base de datos (ID, filas y columnas, datos, etc)
* Interfaz de base de datos
* Mi primer base de datos en **PHPMyAdmin** ([Deploy Fully Managed MySQL in the Cloud](https://www.clever-cloud.com/deploy-cloud-managed-mysql/) )
  + creación de base de datos prueba con tabla (id, nombre, email)
  + incorporar datos
  + tipos de campos

### Práctica

Crear una base de datos en [Deploy Fully Managed MySQL in the Cloud](https://www.clever-cloud.com/deploy-cloud-managed-mysql/) , con la siguiente tabla, agregar tipo de datos e ingresar al menos 3 productos (por ejemplo, notebooks)

* ID
* nombre
* urlImagen
* descripcion
* precio
* stock

### Material Didáctico

* [Bases de datos relacionales vs. no relacionales: ¿qué es mejor? - Aukera](https://aukera.es/blog/bases-de-datos-relacionales-vs-no-relacionales/)
* [Tipos de datos en MySQL para una base de datos SQL](https://disenowebakus.net/tipos-de-datos-mysql.php)
* [MySQL :: MySQL 8.0 Reference Manual :: 11 Data Types](https://dev.mysql.com/doc/refman/8.0/en/data-types.html)
* [40 - Indices de Tablas (Primary Key, Unique, Plain Text, FullText) [MariaDB]](https://www.youtube.com/watch?v=xeEdMbFTgLc)
* [MariaDB vs MySQL, un Resumen sobre las Tecnologías de Base de Datos](https://kinsta.com/es/blog/mariadb-vs-mysql/)
* [Base de datos SQL vs NoSQL: diferencias, ventajas y mitos](https://www.ilimit.com/blog/base-de-datos-sql-nosql/)
* [MySQL](https://docs.google.com/presentation/d/1rbj7utqNGPyF5GpSfCRGF41YpIupQ6y7L7K4nVWnJD4/edit?usp=sharing)
* Hostings de base de datos
  + [Clever Cloud](https://www.clever-cloud.com/deploy-cloud-managed-mysql/)
  + alternativa de host de base de datos: <https://www.alwaysdata.com/en/>

Si existe un error de bloqueo de puertos

* [How to solve "Error: Apache shutdown unexpectedly"? - Stack Overflow](https://stackoverflow.com/questions/18300377/how-to-solve-error-apache-shutdown-unexpectedly)

Clase > <https://youtu.be/-U2d9K8BEpA>

# Clase # 22 - Base de datos parte 2 | 22/5

### Temas

* Sentencias MySQL usando la terminal
* Creación de relación entre 2 tablas mediante terminal

### Práctica

* Hacer ejercicios de [SQLBolt - Learn SQL - SQL Lesson 1: SELECT queries 101](https://sqlbolt.com/lesson/select_queries_introduction)

### Material didáctico

* [Comandos SQL](https://docs.google.com/presentation/d/1iS3SlrXP5ZccELn_ghHi9hoH4c7pKJoGrdFyPEQRZ0s/edit?usp=drive_link)
* [Extensión de VSC](https://marketplace.visualstudio.com/items?itemName=cweijan.vscode-mysql-client2)
* [Consola de mysql online](https://onecompiler.com/mariadb/3zq76m3fn)
* [El lenguaje SQL | Dataprix](https://www.dataprix.com/es/bases-datos-master-software-libre-uoc/lenguaje-sql)
* [CREANDO la BASE DE DATOS de Twitter con MySQL](https://www.youtube.com/watch?v=96s2i-H7e0w)
* [SQL Basics Cheat Sheet | DataCamp](https://www.datacamp.com/cheat-sheet/sql-basics-cheat-sheet)
* [Learn SQL: Queries Cheatsheet | Codecademy](https://www.codecademy.com/learn/learn-sql/modules/learn-sql-queries/cheatsheet)

### 

Clase > <https://youtu.be/Zyj-kTB-QOc>

# Clase # 23 - Base de datos parte 3 | 27/5

### Temas

* Relaciones (1 a 1, 1 a muchos, muchos y muchos)
  + Usando phpMyAdmin
  + Consultas a tablas relacionadas
  + Uso de JOINS, subconsultas, funciones escalares

### Material didáctico

* **Relaciones**
  + [Planificación relaciones](https://fmhelp.filemaker.com/help/18/fmp/es/index.html#page/FMP_Help%2Fplanning-databases.html%23ww1160750)
  + [Relaciones de tablas](https://fmhelp.filemaker.com/help/18/fmp/es/index.html#page/FMP_Help%2Fone-to-one-relationships.html%23)
  + [Relaciones entre tablas de bases de datos – Styde.net](https://styde.net/relaciones-entre-tablas-de-bases-de-datos/)
  + [Uno a muchos | Bases de datos en la enseñanza](http://www.zenodoto.com/gestiondocs/cursosacademicos/manualesrecursos/materalesformaciondocenteTIC/TIC7_BASE/m2_3/uno_a_muchos.html)
  + [SQL para principiantes: Parte 3 - Relaciones de base de datos](https://code.tutsplus.com/es/articles/sql-for-beginners-part-3-database-relationships--net-8561)
  + [mysql - ¿Cómo establecer relación 1:n en base de datos SQL?](https://es.stackoverflow.com/questions/399225/c%C3%B3mo-establecer-relaci%C3%B3n-1n-en-base-de-datos-sql)
  + [🚢 Crear una relación de uno a muchos en MySQL con PhpMyAdmin (CURSO PhpMyAdmin En español 🇪🇸) [06]](https://www.youtube.com/watch?v=xCsT5ef6MTs)
  + [Como Relacionar Tablas en Mysql con phpMyAdmin](https://www.youtube.com/watch?v=5zBb_flFllc)
  + [Cómo unir 3 tablas (o más) en SQL | LearnSQL.es](https://learnsql.es/blog/como-unir-3-tablas-o-mas-en-sql/)
* **Avanzado**
  + [Sentencia JOIN en SQL - Wikipedia, la enciclopedia libre](https://es.wikipedia.org/wiki/Sentencia_JOIN_en_SQL)
  + [¿Qué significa CASCADE, SET NULL, RESTRICT, NO ACTION al crear relaciones?](https://es.stackoverflow.com/questions/105890/qu%C3%A9-significa-cascade-set-null-restrict-no-action-en-mysql)
  + [Oracle Clave primaria vs Clave única vs Índice único - Stack Overflow en español](https://es.stackoverflow.com/questions/28696/oracle-clave-primaria-vs-clave-%C3%BAnica-vs-%C3%8Dndice-%C3%BAnico)
* [MariaDB JOIN - AlphaCodingSkills](https://www.alphacodingskills.com/mariadb/mariadb-join.php)
* [Ejemplos según el PPT de la clase 3 de SQL 2024](https://docs.google.com/document/d/182_zlYDAr2gsiyZnmCrPqAPJqNPZxE2uqNS8z_Y2eMU/edit)

clase > <https://youtu.be/zLBoMOipmno>

# Clase # 24 - Introducción a NodeJS 1 - 29/5

### Temas

* **NodeJS**
  + Instalación
  + Qué es
    - ventajas
  + Hola mundo (ejemplo)
  + Instalación Express usando NPM
  + sitio estático usando Express
    - * usando carpeta public para servir sitio web
      * correr fetch del lado del cliente vs lado servidor

### Práctica

* Recrear el servidor web que se mostró en clase, usando Express.

### Material didáctico

* [Introduction to Node.js - GeeksforGeeks](https://www.geeksforgeeks.org/introduction-to-node-js/)
  + [NodeJS](https://docs.google.com/presentation/d/19o3HTkG5jTv3VROkdZyDkyIh8IpSRcl8gzqoeF4rJ5g/edit#slide=id.p)
  + [¿Por qué demonios usaría Node.js? Un tutorial caso por caso | Toptal](https://www.toptal.com/nodejs/por-que-demonios-usaria-node-js-un-tutorial-caso-por-caso)

**Repaso Asincronismo**

* + [Async in JS: How does it work | Stack Diary](https://stackdiary.com/guides/async-in-js/)
  + [JavaScript Internals: Under The Hood of a Browser | by Vlad Ostrenko | Better Programming](https://betterprogramming.pub/javascript-internals-under-the-hood-of-a-browser-f357378cc922)
  + [Como funciona JavaScript, event loop, call stack, task queue [Descarga PDF gratuita]](https://www.youtube.com/watch?v=kBVPVO0-icw&ab_channel=NavisCode)
    - [Synchronous vs Asynchronous JavaScript – Call Stack, Promises, and More](https://www.freecodecamp.org/news/synchronous-vs-asynchronous-in-javascript/)

**NodeJs - Su funcionamiento técnico**

* [Código bloqueante y no bloqueante con NodeJS | Arsys](https://www.arsys.es/blog/codigo-bloqueante-nodejs)
  + [How Node.js Works | Mosh](https://www.youtube.com/watch?v=jOupHNvDIq8&ab_channel=ProgrammingwithMosh)
  + [Non-blocking I/O and how Node uses it, in friendly terms: blocking vs async IO, CPU vs IO](https://www.youtube.com/watch?v=wB9tIg209-8&ab_channel=StudyingWithAlex)
  + [Node.Js Single Thread Mechanism](https://medium.com/@kelumsampathedirisinghe/node-js-single-thread-mechanism-7a70dc3c2b7f)
  + [How Node.js overcome the problem of blocking of I/O operations ? - GeeksforGeeks](https://www.geeksforgeeks.org/how-node-js-overcome-the-problem-of-blocking-of-i-o-operations/)
  + [If Javascript Is Single Threaded, How Is It Asynchronous? - DEV Community 👩‍💻👨‍💻](https://dev.to/bbarbour/if-javascript-is-single-threaded-how-is-it-asynchronous-56gd)

**Express**

* + [express - npm](https://www.npmjs.com/package/express)
  + [Ejemplo "Hello World" de Express](https://expressjs.com/es/starter/hello-world.html)
  + [Servicio de archivos estáticos en Express](https://expressjs.com/es/starter/static-files.html)

**NPM**

* [#4 - NODE JS - ¿Qué es el package.json? - Introducción práctica PASO A PASO](https://www.youtube.com/watch?v=rAQGFe7lCVs&ab_channel=DorianDesings)
* Licencias: [Licencia de software libre](https://es.wikipedia.org/wiki/Licencia_de_software_libre)
* [¿Que diferencia hay entre NPM y YARN?](https://platzi.com/discusiones/1763-npm/174933-que-diferencia-hay-entre-npm-y-yarn/)
* [¿Tiene sentido Yarn ahora que tenemos npm 5? | campusMVP.es](https://www.campusmvp.es/recursos/post/tiene-sentido-yarn-ahora-que-tenemos-npm-5.aspx)

**Framework para hacer apps de escritorio**

* [Electron | Build cross-platform desktop apps with JavaScript, HTML, and CSS.](https://www.electronjs.org/)

Clase > <https://youtu.be/HbOJF-8Ijhs>

# Clase # 25 - NodeJS 2 - Módulos | 3/6

### Temas

* Métodos de peticiones y códigos de respuesta
* Módulos
  + Nativos (ej: fileSystem)
  + propios
  + Externos (ej: [colors](https://www.npmjs.com/package/colors#usage))
* Desestructuración de objetos
* reinstalar los paquetes al no tener node\_modules
* uso de watch para no reiniciar el servidor

### Material didáctico

**HTTP y métodos**

* [Cabeceras HTTP para Dummies | Envato Tuts+](https://code.tutsplus.com/es/http-headers-for-dummies--net-8039t)
* [Métodos de petición HTTP](https://developer.mozilla.org/es/docs/Web/HTTP/Methods)
* [Códigos de estado de respuesta HTTP](https://developer.mozilla.org/es/docs/Web/HTTP/Status)
* [HTTP Cats](https://http.cat/)
* [Generalidades del protocolo HTTP](https://developer.mozilla.org/es/docs/Web/HTTP/Overview)

**Módulos**

* + [CommonJS vs ES Modules - Javascript en español - Lenguaje JS](https://lenguajejs.com/automatizadores/introduccion/commonjs-vs-es-modules/)
    - [CommonJS vs. ES Modules: Modules and Imports in NodeJS](https://reflectoring.io/nodejs-modules-imports/)
    - [REQUIRE vs IMPORT ⚡ COMMON JS vs ES MODULES 🤔 CJS vs ESM 🟢 Curso de Node.JS desde cero #3](https://www.youtube.com/watch?v=29iYdru2KUg)
    - [The requested module does not provide an export named in JS | bobbyhadz](https://bobbyhadz.com/blog/javascript-requested-module-not-provide-export-named) (errores comunes)
* [FileSystem](https://nodejs.org/docs/latest-v12.x/api/fs.html)
  + otros ejemplos de FS - [Node.js Examples](https://www.tutorialkart.com/nodejs/node-js-examples/)
* [Cómo utilizar la Desestructuración de Arreglos y Objetos en JavaScript](https://www.freecodecamp.org/espanol/news/desestructuracion-de-arreglos-y-objetos-en-javascript/)
* [Módulos en Node.js](https://www.tutorialesprogramacionya.com/javascriptya/nodejsya/detalleconcepto.php?codigo=2&punto=2&inicio=0)

Clase > <https://youtu.be/J5b1vNgiqdE>

# Clase # 26 - NodeJS 3 - Rutas | 5/6

### Temas

* rutas ([Direccionamiento básico de Express](http://expressjs.com/es/starter/basic-routing.html)) (get, put, post, delete)
  + parámetros
    - params (nos permite rutas dinámicas, ej: usuarios/:id)
    - query (URL con variables, ej "?mensaje=hola&costo=1000"
    - body (Para tomar los valores de un formulario)
* Middlewares (su aplicación)
  + Ejemplo de express.urlencoded() y express.json()

### Práctica

* Recrear el crud realizado en clase

### Material didáctico

**Rutas**

* + [Direccionamiento básico de Express](http://expressjs.com/es/starter/basic-routing.html)
  + [Rutas / Routing con NodeJS y Express - Tutorial Stack MEAN](https://funnyfrontend.com/introduccion-al-stack-mean-4-rutas-routing-con-nodejs-y-express/)
  + Manejo de rutas modulando [Direccionamiento de Express](https://expressjs.com/es/guide/routing.html#express-router)
  + [¿Cuáles son las ventajas e inconvenientes de usar PATCH para actualizar recursos?](https://www.linkedin.com/advice/3/what-benefits-drawbacks-using-patch-vs-put-updating?lang=es&originalSubdomain=es#:~:text=Como%20regla%20general%2C%20PATCH%20debe,la%20seguridad%20son%20una%20prioridad) (PATCH vs PUT).

**Parámetros**

* [¿Para qué sirve `app.use(express.json())` en Express?](https://masteringjs.io/tutorials/express/express-json#using-expressjson)  (menciona el uso de req.body)
* [What are express.json() and express.urlencoded()? - Stack Overflow](https://stackoverflow.com/questions/23259168/what-are-express-json-and-express-urlencoded)
* [Método de respuesta JSON en Express.js](https://keepcoding.io/blog/metodo-de-respuesta-json-en-express-js/#:~:text=json()%20analiza%20autom%C3%A1ticamente%20el,body%20).
* [¿Qué podemos hacer con el express.urlencoded?](https://keepcoding.io/blog/que-podemos-hacer-con-el-express-urlencoded/)
* [Parámetros de ruta en express](https://masteringjs.io/tutorials/express/route-parameters)
  + [Parámetro Query en Express](https://masteringjs.io/tutorials/express/query-parameters)
  + [Parámetro Params en Express](https://masteringjs.io/tutorials/express/params)
  + [req.body en Express](https://masteringjs.io/tutorials/express/body#json-request-body)

**Solicitud y respuesta**

* + [¿Qué son los parámetros "res" (respuesta) y "req" (solicitud) en las funciones Express?](https://qastack.mx/programming/4696283/what-are-res-and-req-parameters-in-express-functions)
  + [Códifica Su Primera API con Node.js y Express: Comprensión REST API](https://code.tutsplus.com/es/tutorials/code-your-first-api-with-nodejs-and-express-understanding-rest-apis--cms-31697)

Clase > <https://youtu.be/BoRufBSc8Ck>

# Clase # 27 - NodeJS 4 - Rutas parte 2 | 10/6

### Temas

* CRUD con una DB SQL
* DB
  + Configuración base de datos
  + Pool de conexiones

Paquetes

* [mysql2 - npm](https://www.npmjs.com/package/mysql2)
* [express - npm](https://www.npmjs.com/package/express)

### Material didáctico

**Base de datos**

Paquete mysql2

* [Pool de conexiones - Javascript en el servidor: Node.js](https://manuelmontenegro.github.io/AW-2017-18/04.html#/32/2)
* [MySQL Wildcard Characters](https://www.w3schools.com/mysql/mysql_wildcards.asp)
* [How to filter query results in MySQL](https://www.prisma.io/dataguide/mysql/reading-and-querying-data/filtering-data)

**APIS**

* [¿Qué es una api rest? ¿Cómo funciona? ¿En que tipo de web utilizarlas?](https://dossetenta.com/que-es-una-api-rest/)
* [Ten una REST API Falsa Funcionando Usando json-server | Envato Tuts+](https://code.tutsplus.com/es/fake-rest-api-up-and-running-using-json-server--cms-27871t)
  + Fakes API
    - [dummyJSON - API con información random para testear](https://github.com/Ovi/DummyJSON?tab=readme-ov-file)
    - [JSONplaceholder - API con información random para testear](https://jsonplaceholder.typicode.com/)

Clase > <https://youtu.be/luQK_1qjVI0>

# Clase # 28 - NodeJS 5 | 12/6

### Temas

* Conexión con front para desplegar los productos y tener un buscador
* Creación de filtros

### Práctica

Recrear lo visto en clase para luego poder aplicarlo en el proyecto

### Material didáctico

**Base de datos**

Paquete mysql2

* [How to filter query results in MySQL](https://www.prisma.io/dataguide/mysql/reading-and-querying-data/filtering-data)

**Middlewares**

* [Qué es un Middleware en Express](https://codigofacilito.com/articulos/middleware-express)
* [Middleware en Express JS](https://medium.com/@aarnlpezsosa/middleware-en-express-js-5ef947d668b)
* [Express's json() and urlencoded() Explained - DEV Community](https://dev.to/griffitp12/express-s-json-and-urlencoded-explained-1m7o)
  + [express.json()](https://expressjs.com/es/api.html#express.json)
  + [express.urlencoded](https://expressjs.com/es/api.html#express.urlencoded)
    - [What does 'extended' mean in express 4.0? - node.js](https://stackoverflow.com/questions/29960764/what-does-extended-mean-in-express-4-0)
* [Getting the Request Body in Express - Mastering JS](https://masteringjs.io/tutorials/express/body)

**URL**

* [FormData - Referencia de la API Web | MDN](https://developer.mozilla.org/es/docs/Web/API/FormData)
* [URLSearchParams - Referencia de la API Web | MDN](https://developer.mozilla.org/es/docs/Web/API/URLSearchParams)

Clase > <https://youtu.be/dW4_CXLSL6U>

# Clase # 29 - NodeJS 6 | 19/6

### Temas

* Paquetes
  + multer básico (para subir imágenes al servidor)
* Cómo registrar un producto con una imagen
* Tablas relacionadas (productos y categorias, o por ej cuentas y roles)
* Arquitectura
  + controladores
  + router
  + helper (para manejar los filtros)

### Material didáctico

**Multer**

* [Subir archivos con Multer en Node.js y Express](https://code.tutsplus.com/es/tutorials/file-upload-with-multer-in-node--cms-32088) (hasta la parte de mongo)
* [File Upload Using Multer in Node.js and Express.js](https://www.bacancytechnology.com/blog/file-upload-using-multer-with-nodejs-and-express)
* [multer - npm](https://www.npmjs.com/package/multer)

Clase > <https://youtu.be/XWhnB7smyk8>

# Clase # 30 - NodeJS 7 | 24/6

### Temas

* JWT (JSON Web token)
  + teoría
  + práctica
    - Se puede usar localstorage, pero para mayor seguridad, usamos cookies con [httponly](https://developer.mozilla.org/es/docs/Web/HTTP/Cookies#cookies_secure_y_httponly)

### Material didáctico

**JWT**

* + [JWT: JSON Web Token | ¿Son tan buenos como parecen? 🤔](https://www.youtube.com/watch?v=McMCI0EjMxQ)
  + [¿Qué es JWT y cómo se diferencia de Cookies y Sesiones?](https://programacionymas.com/blog/jwt-vs-cookies-y-sesiones)
  + [JWT tokens and security - working principles and use cases](https://www.vaadata.com/blog/jwt-tokens-and-security-working-principles-and-use-cases/)
    - [XSS (Cross-Site Scripting) vulnerabilities: principles, types of attacks, exploitations and security best practices](https://www.vaadata.com/blog/xss-cross-site-scripting-vulnerabilities-principles-types-of-attacks-exploitations-and-security-best-practices/)
  + [Best Practices for Using JWT](https://blog.bitsrc.io/best-practices-for-using-jwt-df3788433fd3)
  + Bearer
    - [Autenticación con Bearer Token – Documentación API New Bytes](https://developers.nb.com.ar/?page_id=187)
    - [Autenticación y autorización basada en token (JWT Bearer) con ASP.NET Core | Rafael Neto](https://rafaelneto.dev/blog/autenticacion-autorizacion-jwt-bearer-aspnet-core/)
    - [Bearer Authentication](https://swagger.io/docs/specification/authentication/bearer-authentication/)

**JWT Avanzado**

* [10 - JWT (Json web token) - Node Js ¡desde cero!](https://www.youtube.com/watch?v=xr7hv3Qae04) (Link con token)
* [Email Verification - GeeksforGeeks](https://www.geeksforgeeks.org/email-verification/)
* [How to Build A Passwordless Authentication with Email and JWT | HackerNoon](https://hackernoon.com/how-to-build-a-passwordless-authentication-with-email-and-jwt-o33w3311) (verificación de email)

**JWT vs Cookies**

* + [Autenticación con JWT y Cookies 🍪🔐](https://www.youtube.com/watch?v=DxYAcXiy-ak)
  + [Difference between Session Cookies vs. JWT (JSON Web Tokens), for session management.](https://medium.com/@prashantramnyc/difference-between-session-cookies-vs-jwt-json-web-tokens-for-session-management-4be67d2f066e)
  + [Cookies or JWT..? 🤔 - DEV Community](https://dev.to/saurabh2412/cookies-or-jwt-o2i)

Clase > <https://youtu.be/z5uiynUMgLc>

# Clase # 31 - NodeJS 8 | 26/6

### Temas

**Deploy**

* Subir a github
* Subir DB a clever cloud
* Deploy en render

**Seguridad**

* **express-session** para mantener la sesión de login abierta
* **bcrypt** para almacenar contraseñas seguras.

### Material didáctico

**Deploy**  
 Durante el deploy, en por ej, render o etc, si aparece el mensaje de advertencia: *Warning: connect.session() MemoryStore is not designed for a production environment, as it will leak memory, and will not scale past a single process.*

* [La explicación de ese mensaje](https://medium.com/developer-rants/how-to-handle-sessions-properly-in-express-js-with-heroku-c35ea8c0e500)
* **Express-session**
  + Propiedades necesarias
    - saveUninitialized [When to use saveUninitialized and resave in express-session - Stack Overflow](https://stackoverflow.com/questions/40381401/when-to-use-saveuninitialized-and-resave-in-express-session) | ([en español](https://stackoverflow-com.translate.goog/questions/40381401/when-to-use-saveuninitialized-and-resave-in-express-session?_x_tr_sl=auto&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=wapp))
    - resave: [user-content-resave](https://www.npmjs.com/package/express-session#user-content-resave)
    - secret: [Secure Session](https://blog.jscrambler.com/best-practices-for-secure-session-management-in-node/)
  + [Introducción a Cookies y Sesiones con Node.js](https://appdelante.com/cursos/cookies-y-sesiones-con-nodejs/configuracion-del-modulo-express-session)
  + [How to Manage Session using Node.js and Express | CodeForGeek](https://codeforgeek.com/manage-session-using-node-js-express-4/)
  + [Cookie and Session (I): The basic concept | by Alysa Chan | Medium](https://medium.com/@alysachan830/cookie-and-session-i-the-basic-concept-fea3d52bc8d3)
  + [Cookie and Session (II): How session works in express-session | by Alysa Chan | Medium](https://medium.com/@alysachan830/cookie-and-session-ii-how-session-works-in-express-session-7e08d102deb8)
  + [Which is Better Node.js Session Management Libraries? cookie-parser vs express-session vs cookie-session](https://npm-compare.com/cookie-parser,cookie-session,express-session)
* **Bcrypt**
  + [Encriptar contraseñas con Node - Parzibyte's blog](https://parzibyte.me/blog/2020/08/13/encriptar-contrasenas-node/)
  + <https://www.itsecurityguru.org/wp-content/uploads/2023/11/Time-to-Crack-bcrypt-hashed-passwords.jpg>
  + [User Authentication and Authorization: Password Authentication Cheatsheet | Codecademy](https://www.codecademy.com/learn/becj-22-user-authentication-and-authorization/modules/wdcp-22-password-authentication-e72aec5b-6de4-4d9c-9a91-5947bfb0c01a/cheatsheet)
  + [How to securely store passwords? - DEV Community](https://dev.to/techschoolguru/how-to-securely-store-passwords-3cg7)
  + [bcrypt - npm](https://www.npmjs.com/package/bcrypt)

Clase > <https://youtu.be/hhO7NgIg7QY>

# Clase # 32 - Presentación de trabajos #1 | 1/7

### Temas

* Presentación del trabajo de los equipos

Clase > <https://youtu.be/FvcrXWIxkf4>

# Clase # 33 - Presentación de trabajos #2 | 3/7

### Temas

* Presentación del trabajo de los equipos

Clase > <https://youtu.be/43g9gkjMEYw>

# Clase # 34 - Presentación de trabajos #3 | 8/7

### Temas

* Presentación del trabajo de los equipos

Clase > <https://youtu.be/N6virz3Bao0>

# Clase # 35 - Cierre | 10/7

### Temas

* Últimas presentaciones del trabajo de los equipos
* Futuro: ¿ahora qué?

### Material

Freelancers

* [DEMO Presupuesto - template](https://docs.google.com/document/d/1n4Dm-rZu4p913wCYxhrQDjc-s3C3j78JYPCBsfdZCxg/edit)

**Cobrar**

* [Tarifario.org](https://tarifario.org)
* [Valor hora / Tarifas mínimas - ADCV](https://www.adg.com.ar/tarifario2/)
* <https://www.instagram.com/p/CeFBkTGuF4c/> | Formas de trato al cliente
* [¿Puedo cobrar servicios que presté en el exterior en mi cuenta en dólares?](https://ayuda.bancogalicia.com.ar/n4/puedo-cobrar-servicios-que-preste-en-el-exterior-en-mi-cuenta-en-dolares)
* [¿Cuánto gana un PROGRAMADOR JUNIOR en ARGENTINA?💸Tips para comenzar 💻](https://www.youtube.com/watch?v=olu4JzEc-mU)

**Encontrar trabajo como freelance**

* [Las 8 mejores plataformas para contratar freelance – 2024](https://www.websiteplanet.com/es/freelance-websites/)
* [Freelancer: ¿Qué es y cómo usarlo para encontrar empleo online?](https://trabajarporelmundo.org/freelancer-empleo-online/)
* [Fiverr](https://es.fiverr.com/)
* [Freelancer](https://www.freelancer.com.ar/)
* [Workana](https://www.workana.com/es/)
* [¿Tienes pensado buscar TRABAJO REMOTO en PROGRAMACIÓN?](https://x.com/midudev/status/1556350784157138944?s=20)
* [7 WEBS para encontrar trabajo REMOTO en PROGRAMACIÓN!](https://x.com/midudev/status/1708828644750098567?s=20)

**Datos**

* [Infografía sobre el estado del ecosistema de los desarrolladores en 2023 | JetBrains: Developer Tools for Professionals and Teams](https://www.jetbrains.com/es-es/lp/devecosystem-2023/)
* [Encuesta sueldos 2024](https://sueldos.openqube.io/encuesta-sueldos-2024.01/)  (Argentina)

**Búsquedas**

* <https://roadmap.sh/roadmaps>
* [Búsqueda "Desarrollador front end"](https://www.linkedin.com/jobs/search/?currentJobId=3962826105&keywords=Desarrollador%20de%20front-end&origin=JOBS_HOME_SEARCH_BUTTON&refresh=true)
* [Programador full stack](https://www.linkedin.com/jobs/search/?currentJobId=3959740114&geoId=100446943&keywords=Programador%20full%20stack&origin=JOB_SEARCH_PAGE_KEYWORD_AUTOCOMPLETE&refresh=true)
* Entrevistas <https://www.youtube.com/results?search_query=preparacion+entrevista+trabajo>

**Roadmaps**

* [Step by step guide to learn JavaScript](https://roadmap.sh/javascript)
* [What is Frontend Development?](https://roadmap.sh/frontend)
  + - [Learn to become a React developer](https://roadmap.sh/react)
* [What is Backend Development?](https://roadmap.sh/backend)
  + - [Learn to become a modern node.js developer](https://roadmap.sh/nodejs)

**Plataformas donde seguir estudiando**

* [https://www.codecademy.com/](https://www.codecademy.com/es) (en inglés) [(nota)](http://www.redusers.com/noticias/entrevista-codecademy-programacion-escuelas/)
* <https://learn.freecodecamp.org> (en inglés) [(nota)](https://www.genbeta.com/web/aprende-a-programar-gratis-en-freecodecamp-y-gana-experiencia-escribiendo-codigo-para-organizaciones-sin-fines-de-lucro)

clase > <https://youtu.be/6R6g-rSAuLs>