

¿Qué sabemos ya?

Gestionar versiones del SW con git

Subir una aplicación a Heroku

Qué es Heroku

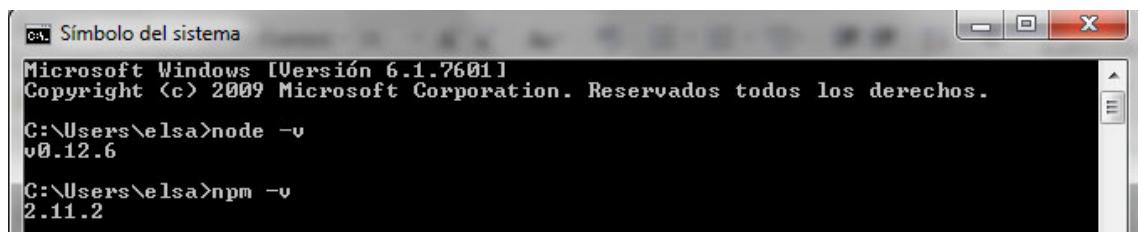
¿Qué no sabemos aún?

Cómo programar un servidor Web para desplegarlo en Heroku:

- Node.js: permite ejecutar JS en el servidor

Instalar node.js y npm (<https://nodejs.org/en/download/>)

Verificar instalación:



```
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\elsa>node -v
v0.12.6

C:\Users\elsa>npm -v
2.11.2
```

- Módulos que se instalan con npm como express, pg, cool, ... (son librerías de JS, conjunto de funciones que queremos incluir en nuestra aplicación)

```
var miModulo= require('pg');
```

```
npm install pg --save
```

- Manejo de vistas con ejs (Embedded JS): genera código HTML a partir de datos que se le pasan a una plantilla

```
<!DOCTYPE html>
<html>
<head>
  <% include ../partials/header.ejs %>
</head>

<body>

  <% include ../partials/nav.ejs %>

  <h2>Database Results</h2>

  <ul>
    <% results.forEach(function(r)  %>
      <li><%= r.id %> - <%= r.name %></li>
    <% } ); %>
  </ul>

</body>
</html>
```

- Archivos de configuración: Procfile y package.json

Punto de entrada a la aplicación (proceso que debe ejecutar)

web: node index.js

Heroku reconoce una aplicación de tipo Node.js si encuentra en el directorio raíz el archivo package.json.

PASO 1: Crear un esqueleto de la aplicación usando la herramienta [express-generator](#)

Tutorial en <http://expressjs.com/en/starter/generator.html>.

En una ventana de órdenes, dentro del directorio donde queramos empezar a programar nuestro servidor, ejecutar las siguientes órdenes (ir tomando captura de lo que va sucediendo en cada resultado de la ejecución de la orden correspondiente):

```
npx express-generator -- view=ejs myapp
```

```
C:\ELSA\DOCENCIA\TING>npx express-generator --view=ejs myapp
```

Este equipo > Windows (C:) > ELSA > DOCENCIA > TING > myapp			
Nombre	Fecha de modificación	Tipo	T
bin	06/10/2020 11:29	Carpeta de archivos	
public	06/10/2020 11:29	Carpeta de archivos	
routes	06/10/2020 11:29	Carpeta de archivos	
views	06/10/2020 11:29	Carpeta de archivos	
app	06/10/2020 11:29	Archivo JavaScript	
package	06/10/2020 11:29	Archivo JSON	

```
cd myapp
```

```
npm install
```

```
C:\ELSA\DOCENCIA\TING>cd myapp
```

```
C:\ELSA\DOCENCIA\TING\myapp>npm install
```

```
C:\ELSA\DOCENCIA\TING\myapp>npm install
npm notice created a lockfile as package-lock.json. You should commit this file.
added 54 packages from 38 contributors and audited 55 packages in 2.682s
found 0 vulnerabilities
```

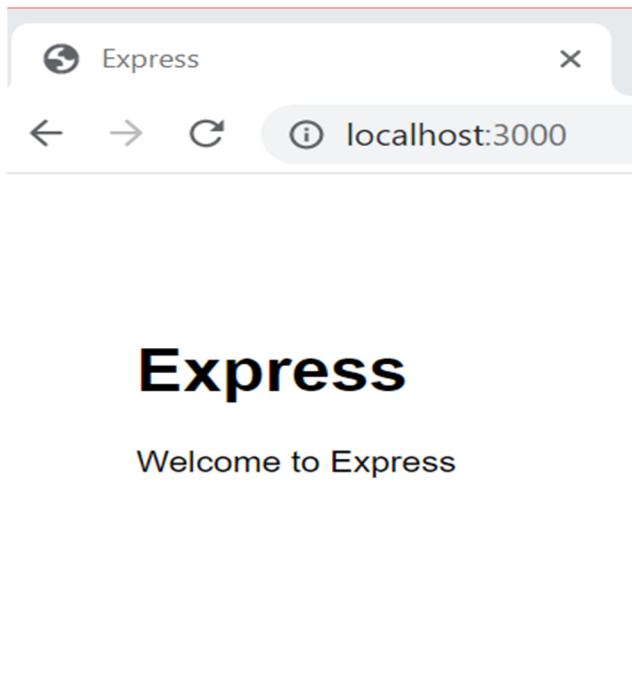
Este equipo > Windows (C:) > ELSA > DOCENCIA > TING > myapp

Nombre	Fecha de modificación
bin	06/10/2020 11:29
node_modules	06/10/2020 11:34
public	06/10/2020 11:29
routes	06/10/2020 11:29
views	06/10/2020 11:29
app	06/10/2020 11:29
package	06/10/2020 11:29
package-lock	06/10/2020 11:34

npm start

```
C:\ELSA\DOCENCIA\TING\myapp>npm start
```

```
> myapp@0.0.0 start C:\ELSA\DOCENCIA\TING\myapp
> node ./bin/www
```



PASO 2: subir esta aplicación a Heroku

Crear y editar archivo Procfile → copiar del proyecto clonado “node-js-getting-started” y adaptar su contenido: web: node ./bin/www

Crear y editar archivo .gitignore (opcional) → lo mismo que para Procfile, podemos copiar y modificar su contenido (en principio, podemos dejarlo como está, al menos que ignore la carpeta node_modules)

Este equipo > Windows (C:) > ELSA > DOCENCIA > TING > myapp

Nombre	Fecha de modificación
bin	06/10/2020 11:29
node_modules	06/10/2020 11:34
public	06/10/2020 11:29
routes	06/10/2020 11:29
views	06/10/2020 11:29
.gitignore	02/10/2020 10:22
app	06/10/2020 11:29
package	06/10/2020 11:29
package-lock	06/10/2020 11:34
Procfile	11/10/2020 9:03

Crear una confirmación (commit con git)

```
C:\ELSA\DOCENCIA\TING\myapp>git init
Initialized empty Git repository in C:/ELSA/DOCENCIA/TING/myapp/.git/

C:\ELSA\DOCENCIA\TING\myapp>git add .
warning: LF will be replaced by CRLF in app.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in bin/www.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in package-lock.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in package.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in public/stylesheets/style.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in routes/index.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in routes/users.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in views/error.ejs.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in views/index.ejs.
The file will have its original line endings in your working directory
```

```
C:\ELSA\DOCENCIA\TING\myapp>git commit -m "mi primera App en Heroku version 1"
[master (root-commit) f59415f] mi primera App en Heroku version 1
Committer: Elsa María Macías López <elsa.macias@ulpgc.es>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

11 files changed, 604 insertions(+)
create mode 100644 .gitignore
create mode 100644 Procfile
create mode 100644 app.js
create mode 100644 bin/www
create mode 100644 package-lock.json
create mode 100644 package.json
create mode 100644 public/stylesheets/style.css
create mode 100644 routes/index.js
create mode 100644 routes/users.js
create mode 100644 views/error.ejs
create mode 100644 views/index.ejs
```

Enviar la app a la nube de Heroku (heroku create y push heroku master)

Esto es lo que sucede cuando has superado la cantidad máxima de apps que puedes desplegar (hasta 5):

```
C:\ELSA\DOCENCIA\TING\myapp>heroku create
Creating app... !
! You've reached the app limit of 5 apps for unverified accounts. Delete some apps or add a credit card to verify
! your account.
```

Una vez eliminada, puedo crear la app:

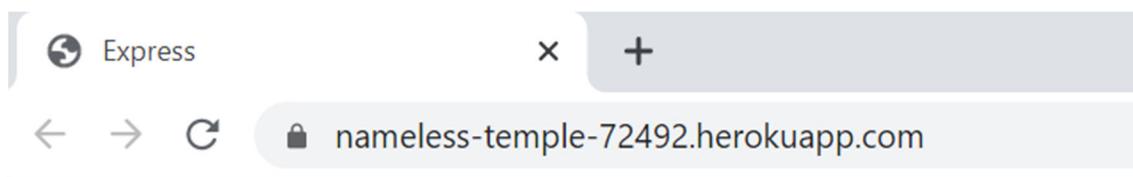
```
C:\ELSA\DOCENCIA\TING\myapp>heroku create
Creating app... done, ⚡ nameless-temple-72492
https://nameless-temple-72492.herokuapp.com/ | https://git.heroku.com/nameless-temple-72492.git
```

```
C:\ELSA\DOCENCIA\TING\myapp>git push heroku master
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 8 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (18/18), 7.27 KiB | 1.82 MiB/s, done.
Total 18 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Node.js app detected
remote:
remote: -----> Creating runtime environment
remote:
remote:           NPM_CONFIG_LOGLEVEL=error
remote:           NODE_ENV=production
remote:           NODE_MODULES_CACHE=true
remote:           NODE_VERBOSE=false
remote:
remote: -----> Installing binaries
remote:           engines.node (package.json): unspecified
remote:           engines.npm (package.json): unspecified (use default)
remote:
remote:           Resolving node version 12.x...
remote:           Downloading and installing node 12.19.0...
remote:           Using default npm version: 6.14.8
remote:
remote: -----> Installing dependencies
remote:           Installing node modules
remote:           added 54 packages in 1.115s
remote:
remote: -----> Build
remote:
remote: -----> Caching build
remote:           - node_modules
remote:
remote: -----> Pruning devDependencies
```

```
remote:      audited 55 packages in 0.764s
remote:      found 0 vulnerabilities
remote:
remote:
remote: -----> Build succeeded!
remote: -----> Discovering process types
remote:           Procfile declares types -> web
remote:
remote: -----> Compressing...
remote:           Done: 22.9M
remote: -----> Launching...
remote:           Released v3
remote:           https://nameless-temple-72492.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/nameless-temple-72492.git
 * [new branch]      master -> master
```

Lanzar la app: heroku open

```
C:\ELSA\DOCENCIA\TING\myapp>heroku open
```



Express

Welcome to Express

Paso 3: Modificar código para cambiar alguna vista

/bin/www lanza app.js en el puerto local 3000. La aplicación tiene 2 rutas (paths) programadas:

1. / → ejecuta el archivo index.js que visualiza la vista de index.ejs (método render)

2. /users → ejecuta el archivo users.js que envía un mensaje a visualizar en el navegador (método send)

Por ejemplo, para obtener este resultado

Express

Welcome to Express

TING

Welcome to TING

Soy la alumna

Elsa

de apellido

Macías

Tenemos que cambiar en los archivos index.js e index.ejs

index.js

 index: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
var express = require('express');
var router = express.Router();
```

```
/* GET home page. */
```

```
router.get('/', function(req, res, next) {
  res.render('index', { title: 'Express', title2:'TING', title3: 'Elsa', title4:'Macías' });
});
```

```
module.exports = router;
```

index.ejs

 index: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title><%= title %></title>
```

```
    <link rel='stylesheet' href='/stylesheets/style.css' />
```

```
  </head>
```

```
  <body>
```

```
    <h1><%= title %></h1>
```

```
    <p>Welcome to <%= title %></p>
```

```
    <h2><%= title2 %></h2>
```

```
    <p>Welcome to <%= title2 %></p>
```

```
    <h3> Soy la alumna </h3>
```

```
    <p> <%= title3 %> </p>
```

```
    <h3> de apellido </h3>
```

```
    <p><%= title4 %> </p>
```

```
  </body>
```

```
</html>
```