



El futuro digital  
es de todos

MinTIC

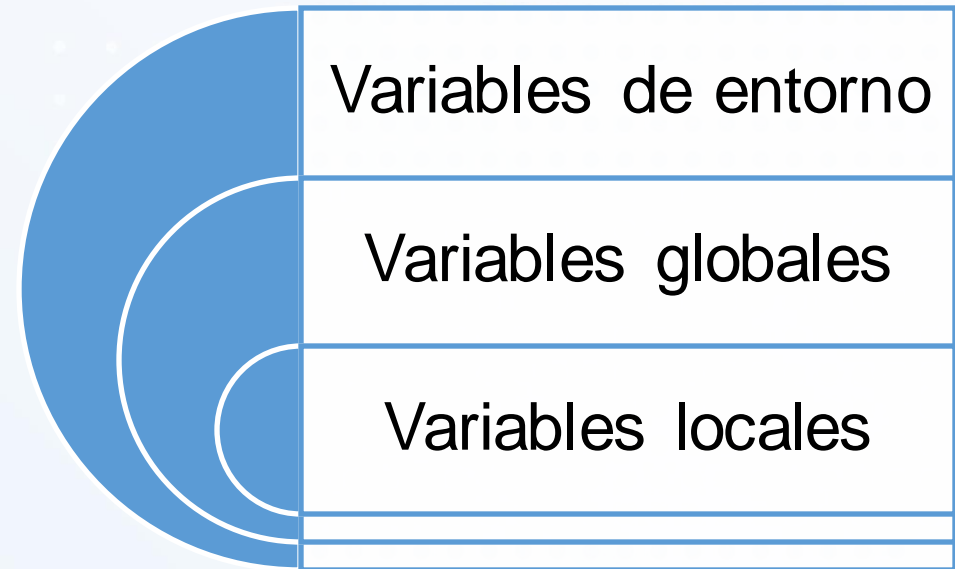
# Capacitación de aplicaciones web – Publicando la App

Por equipo de tutores



# Preparando backend Node para despliegue

En cualquier aplicación es importante declarar las variables de entorno, estas variables no dependen de un script o clase, además pueden ser globales y existen en toda la aplicación



# Variables de entorno

1)

```
>>> npm i dotenv
```

2) Creamos un archivo llamado .env que contiene:

```
PORT=3000
```

3) Usamos la variable de entorno para express, ej:

```
require('dotenv').config()
const express = require('express')
const app = express()
app.get('/', function (req, res) {
  res.send('Hello World')
})
app.listen(process.env.PORT)
```

# Archivos estáticos

- En express podemos configurar un middleware que permita añadir archivos estáticos contenidos en una carpeta llamada public (puede ser cualquier nombre)

```
app.use( express.static('public') );
```

# Crear un HTML para la ruta comodín

- Para cualquier ruta no definida se define la ruta comodín como: '\*'

```
path = require('path');  
app.get('*', (req, res) => { res.sendFile(  
  path.resolve('./public/index.html') ) })
```

- En este caso, el controlador responde con un index que se encuentra en la carpeta de public, la función resolve solo requiere la ruta relativa del archivo index.html



# CORS

(Intercambio de recursos de origen cruzado)  
Nos permiten decidir quien puede solicitar recursos a nuestro Backend

```
>>> npm i cors
```

Usamos un middleware de cors

```
const cors = require('cors');  
app.use( cors() );
```

# COMANDO START

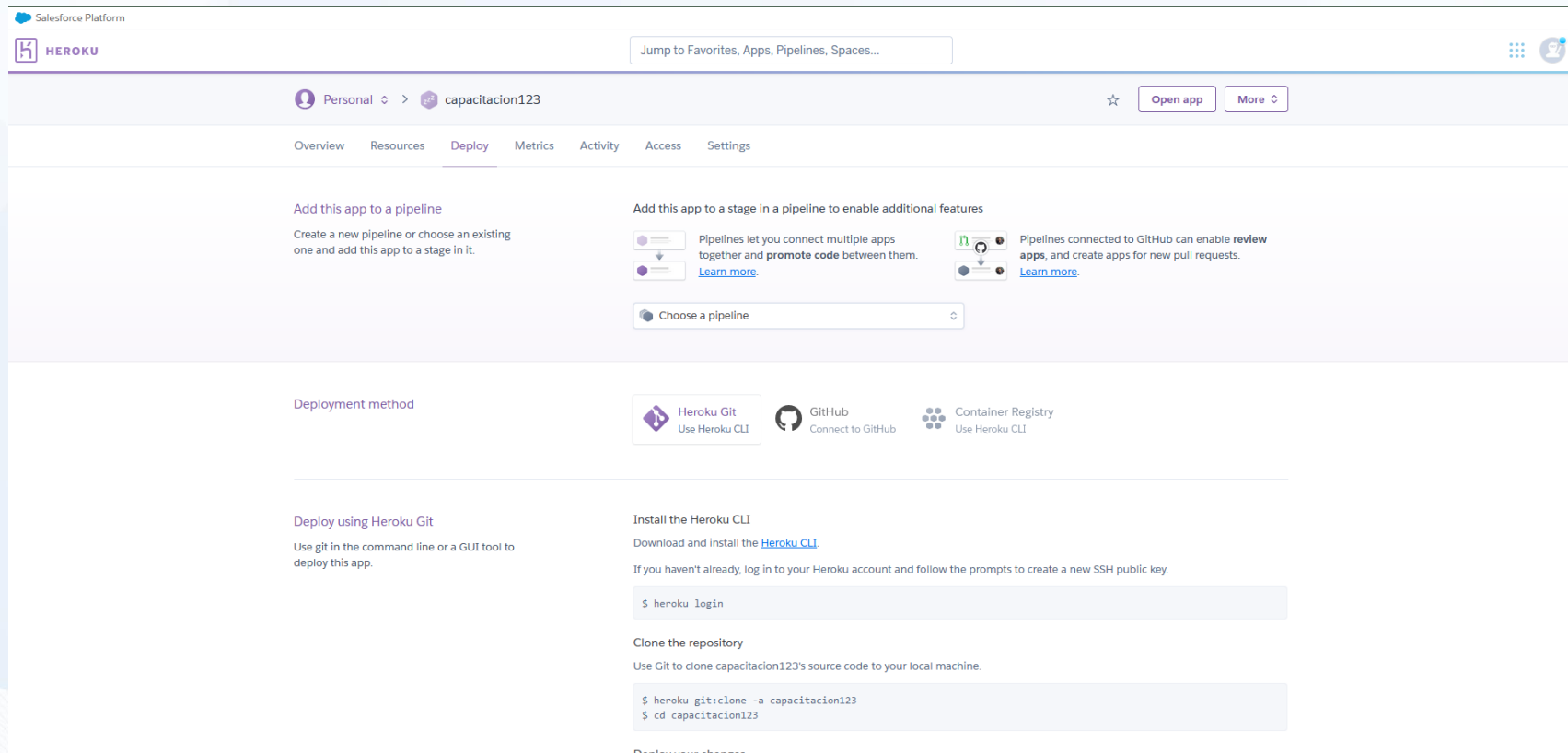
En package.json agregamos un comando de npm que va ejecutar Heroku automáticamente :

```
"scripts": {  
  "start": "node index"  
},
```

```
>>> npm run start
```

# HEROKU

Creamos la aplicación para el proyecto en: new / create new app  
Después de crear la app, nos ubicamos en el dashboard:





# HEROKU

Nuestro proyecto debe ser un repositorio de GIT, luego:

Seguimos los pasos usando Heroku Git

## Deployment method



Heroku Git  
Use Heroku CLI



GitHub  
Connect to GitHub



Container Registry  
Use Heroku CLI

## Deploy using Heroku Git

Use git in the command line or a GUI tool to deploy this app.

### Install the Heroku CLI

Download and install the [Heroku CLI](#).

If you haven't already, log in to your Heroku account and follow the prompts to create a new SSH public key.

```
$ heroku login
```

### Clone the repository

Use Git to clone capacitacion123's source code to your local machine.

```
$ heroku git:clone -a capacitacion123  
$ cd capacitacion123
```

### Deploy your changes

Make some changes to the code you just cloned and deploy them to Heroku using Git.

```
$ git add .  
$ git commit -am "make it better"  
$ git push heroku master
```



You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

# Conexión Base de Datos - MongoDB

CLUSTERS > CREATE A SHARED CLUSTER

## Create a Shared Cluster

Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our [documentation](#).

PREVIEW Serverless

Dedicated

FREE Shared

For learning and exploring MongoDB in a sandbox environment. Basic configuration controls.

No credit card required to start. Upgrade to dedicated clusters for full functionality.  
Explore with sample datasets. Limit of one free cluster per project.

### 1 How would you like to authenticate your connection?

Your first user will have permission to read and write any data in your project.

Username and Password

Certificate

Create a database user using a username and password. Users will be given the *read and write to any database* privilege by default. You can update these permissions and/or create additional users later. Ensure these credentials are different to your MongoDB Cloud username and password.

Username

Password

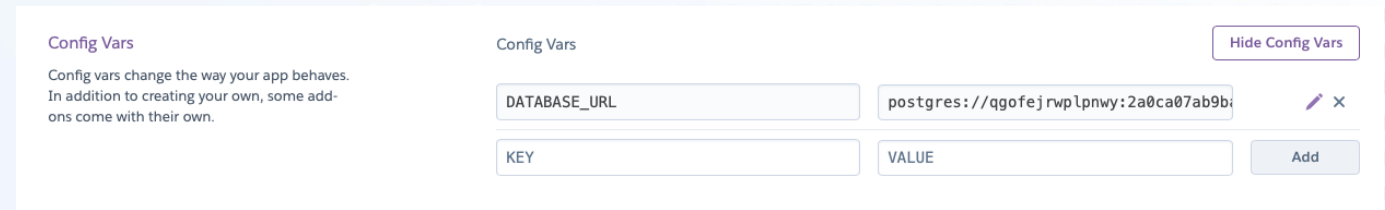
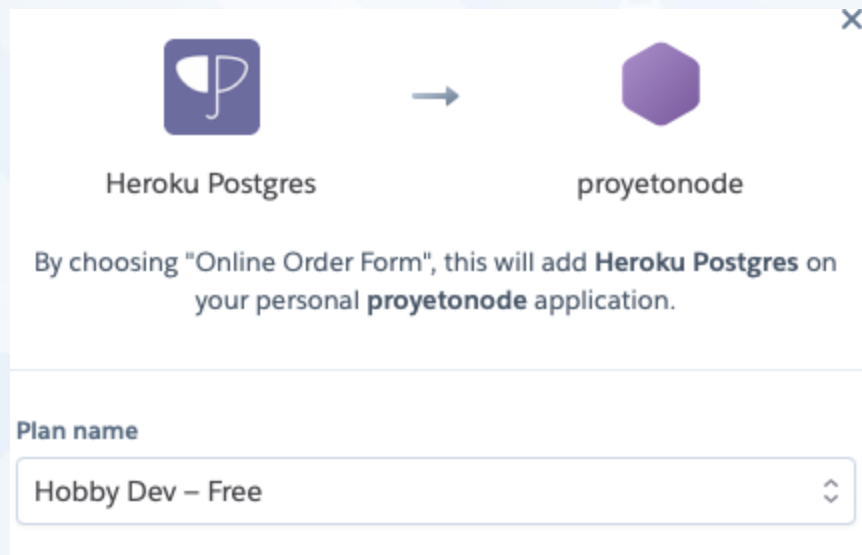
misiontic

misiontic

Create User

Se crea una base de datos en MongoDB Atlas , donde mediante especificando la IP de Heroku hacemos acceso a esta misma para su respectiva administración.

# Conexión Base de Datos - Postgres

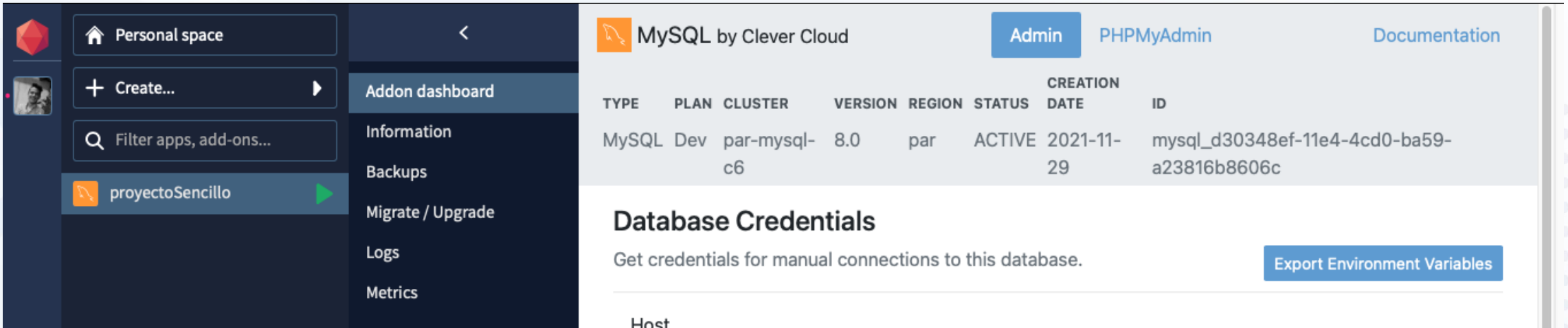


Se crea el add-on de Heroku-Postgres (Plan Hobby Dev) y en la conexión de la base de datos cambiamos las claves de acceso , de acuerdo a como las provee Heroku (Settings-Config Vars)

# Conexión Base de Datos - Postgres

postgres://usuario:contrasena@host:puerto/basededatos

# Conexión Base de Datos - MySQL



The screenshot displays the MySQL by Clever Cloud dashboard. On the left is a sidebar with navigation options: Personal space, Create..., Filter apps, add-ons..., proyectoSencillo (selected), Addon dashboard, Information, Backups, Migrate / Upgrade, Logs, and Metrics. The main content area shows the MySQL instance details in a table and a section for database credentials.

TYPE	PLAN	CLUSTER	VERSION	REGION	STATUS	CREATION DATE	ID
MySQL	Dev	par-mysql-c6	8.0	par	ACTIVE	2021-11-29	mysql_d30348ef-11e4-4cd0-ba59-a23816b8606c

**Database Credentials**  
Get credentials for manual connections to this database.

Host

Buttons: Admin, PHPMyAdmin, Documentation, Export Environment Variables

Mediante la utilización de Clever Cloud , podemos crear una base de datos sencilla MySQL , la cual nos dará la respectiva información a utilizar para conectar nuestro back-end al proyecto.



# LocalStorage

LocalStorage nos permite persistir información en el navegador, información la cual debe guardarse como string

```
localStorage.setItem('miGato', 'Juan');
```

**Nota:** Por favor ver el artículo [Usando la Web Storage API](#) para un ejemplo completo.

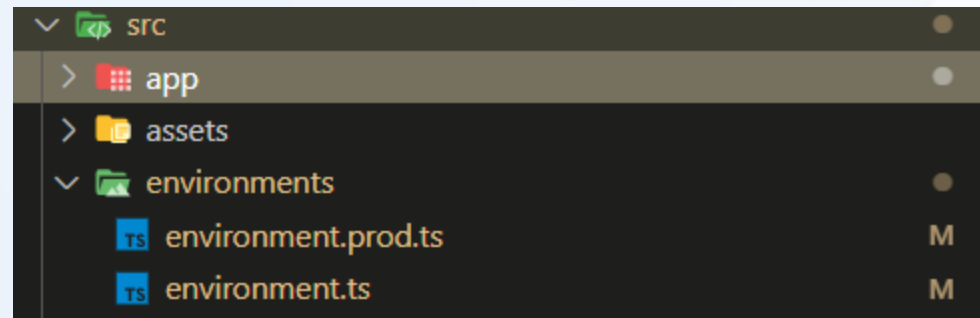
La sintaxis para leer el ítem almacenado en `localStorage` es la siguiente:

```
var cat = localStorage.getItem('miGato');
```

# Variables de entorno Angular

```
environment.ts M x
src > environments > environment.ts > ...
1 // This file can be replaced during build
2 // `ng build` replaces `environment.ts`
3 // The list of file replacements can be
4
5 export const environment = {
6   production: false,
7   base_url: 'http://localhost:3000/api/'
8 };
9

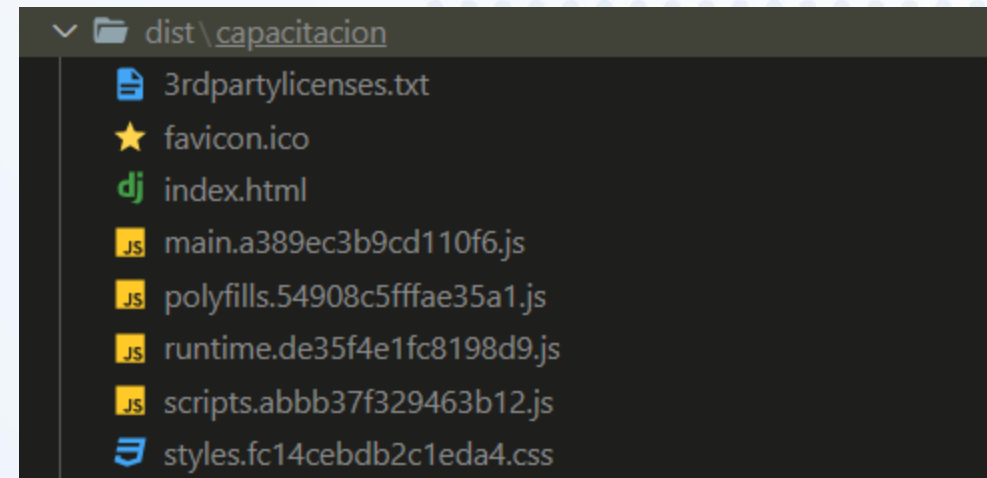
environment.prod.ts M x
src > environments > environment.prod.ts > ...
1 export const environment = {
2   production: true,
3   base_url: 'https://capacitacion123.herokuapp.com/'
4 };
5
```



# Compilar Angular

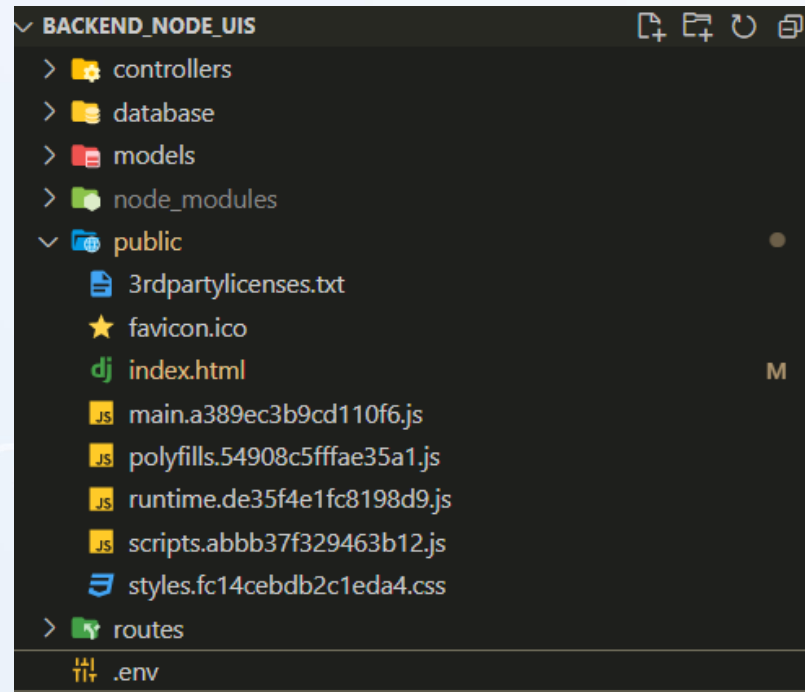
Angular compila el proyecto en archivos estaticos ejecutando:

```
>>> ng build
```



# Express server y Angular

Express puede servir archivos estaticos, con las configuraciones ya indicadas podemos copiar angular construido en la carpeta de public. Express se encarga de ejecutar Angular haciendo uso de la ruta comodin



# Publicar Frontend en vercel

Vercel es una plataforma para **marcos frontend y sitios estáticos** , creada para integrarse con su contenido.

Brinda una **experiencia de desarrollador sin fricciones** para encargarse de las cosas difíciles: implementar instantáneamente, escalar automáticamente y ofrecer contenido personalizado en todo el mundo.


Hace que sea fácil para los equipos de frontend **desarrollar, obtener una vista previa y publicar.**





# Publicar Frontend en vercel

1: Primero registrate en: <https://vercel.com/login> con tu cuenta de GitHub preferiblemente

## Log in to Vercel

 Continue with GitHub

 Continue with GitLab

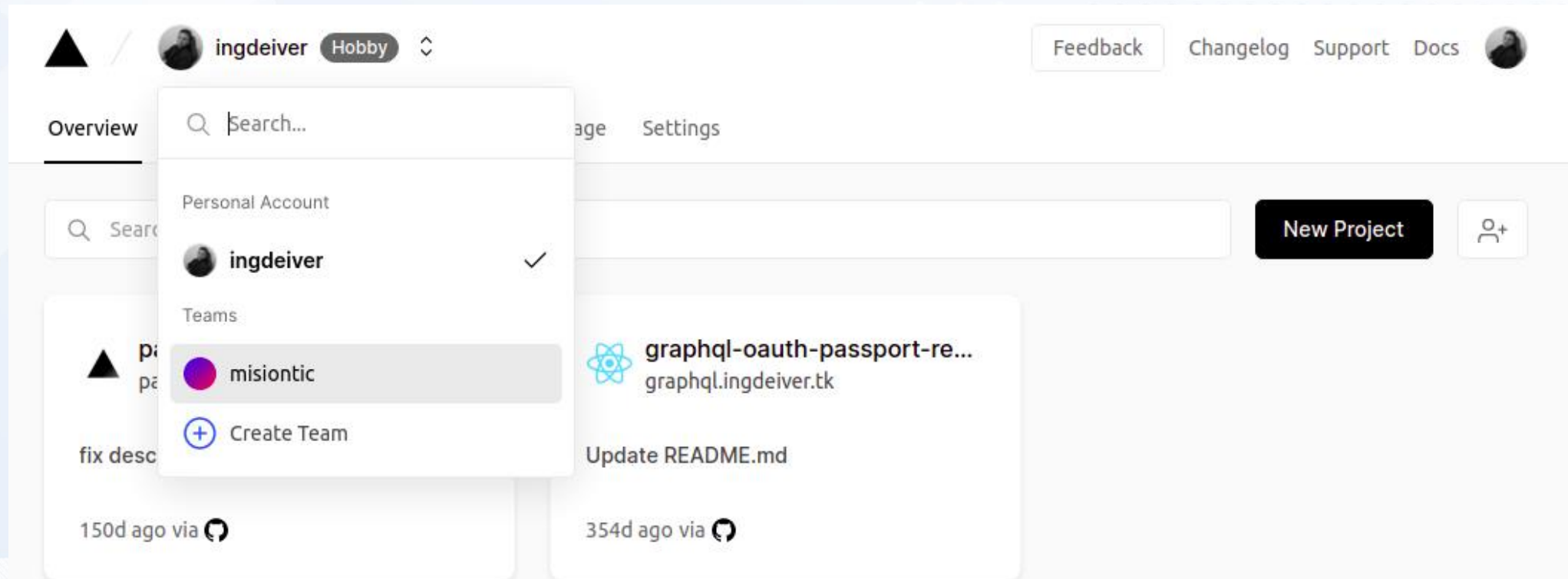
 Continue with Bitbucket

 Continue with SAML SSO

[Continue with Email →](#)

# Publicar Frontend en vercel

**2:** Seleccciona o crea un equipo, un equipo se usa para organizar los proyectos, por ejemplo un equipo para proyectos de misiontic





# Publicar Frontend en vercel





3: Haz clic en **New project.**

4: **Importa tu repositorio de GitHub,** recuerda darle los permisos necesarios a vercel para que puede acceder a todos los repositorios o solo a algunos de ellos.

### Import Git Repository

 IngDeiver ▼

 Search...

	graphql-oauth-passport... · 341d ago	<a href="#">Import</a>
	misiontic · 4h ago	<a href="#">Import</a>
	next-mercadopago-mui · 150d ago	<a href="#">Import</a>
	portafolio · 119d ago	<a href="#">Import</a>

Missing Git repository? [Adjust GitHub App Permissions](#) →

# Publicar Frontend en vercel

**5: Selecciona el framework** que usaste para tu app frontend y Vercel se encargara de toda la configuración necesaria

**6: Llena la información basica del proyecto, haz clic en **Deploy** y espera a que se compile**

The screenshot shows the Vercel 'Configure Project' interface. On the left, a sidebar displays the project name 'misiontic' and a navigation menu with 'Configure Project' (selected) and 'Deploy'. Below the menu, the 'GIT REPOSITORY' section shows 'IngDeiver/misiontic' with a 'main' branch and a './' directory. At the bottom of the sidebar is a link to 'Import a different Git Repository'. The main area is titled 'Configure Project' and contains three input fields: 'PROJECT NAME' (filled with 'misiontic'), 'FRAMEWORK PRESET' (a dropdown menu showing 'Angular'), and 'ROOT DIRECTORY' (filled with './' and an 'Edit' button). At the bottom of the main area are two expandable sections: 'Build and Output Settings' and 'Environment Variables'.

# Publicar Frontend en vercel

## Practica web con Angular

### Crud de libros

<input type="text" value="Titulo"/>	<input type="text" value="Categoría"/>
<input type="text" value="Numero de páginas"/>	<input type="text" value="ISBN"/>
<input type="text" value="Autor"/>	<input type="text" value="Editorial"/>

Guardar

#	Título	Categoría	N. páginas	ISBN	Autor	Editorial	Acción
example works!							

Go to Dashboard

DEVELOP

Run `ng serve` to run your project locally

PREVIEW

Push to any Git branch other than `main` to preview changes

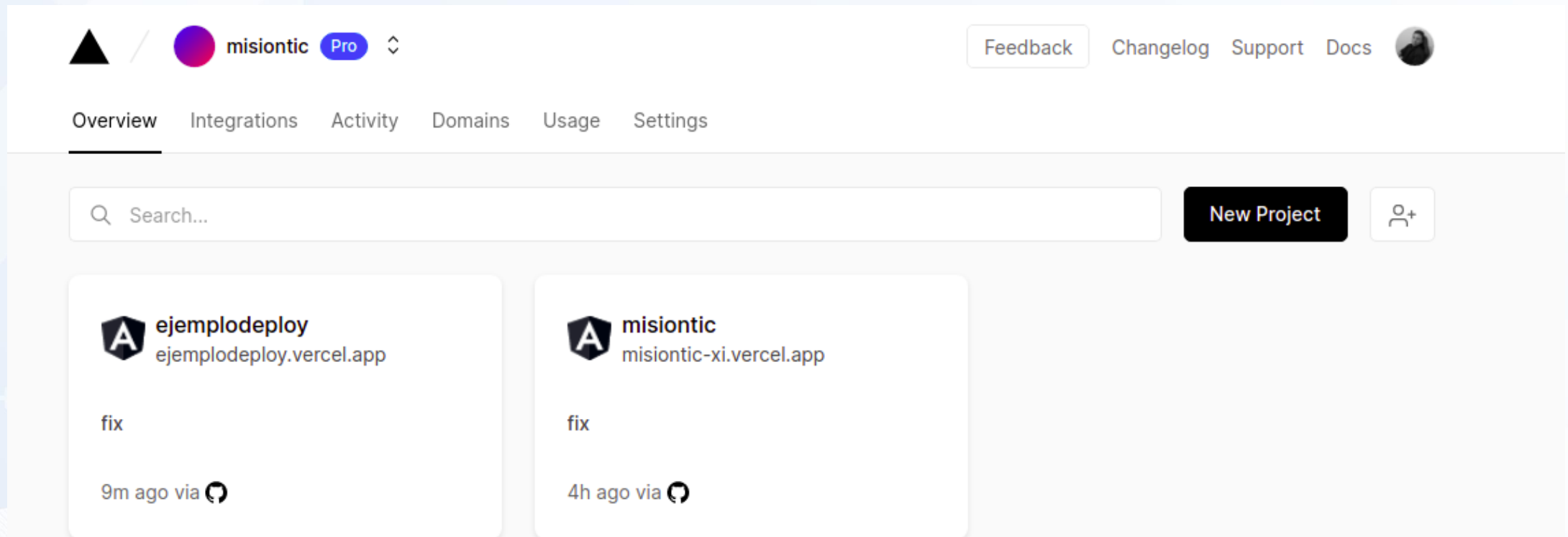
SHIP

Push to `main` to ship changes to production



# Publicar Frontend en vercel

En tu **Dashboard** aparecerán los equipos que haz creado, selecciona el quipo que creates y en se listaran todos los proyectos que haz creado/publicado, si haces clic sobre alguno podras ver información del proyecto, como el dominio y una vista previa



The screenshot displays the Vercel dashboard interface. At the top, the user 'misiontic' is logged in with a 'Pro' plan. Navigation links include 'Feedback', 'Changelog', 'Support', 'Docs', and a user profile icon. Below this, a menu bar shows 'Overview' (selected), 'Integrations', 'Activity', 'Domains', 'Usage', and 'Settings'. A search bar and a 'New Project' button are also visible. The main content area lists two projects:

- ejemplodeploy**: ejemplodeploy.vercel.app, with a 'fix' deployment 9m ago via GitHub.
- misiontic**: misiontic-xi.vercel.app, with a 'fix' deployment 4h ago via GitHub.

# Publicar Frontend en vercel

En tu **Dashboard** aparecerán los equipos que haz creado, selecciona el quipo que creates y se listaran todos los proyectos que haz creado, si haces clic sobre alguno podras ver información del proyecto, como el dominio y una vist previa

The screenshot shows the Vercel dashboard interface. At the top, there's a navigation bar with the Vercel logo, a team selector (currently showing 'misiontic' with a 'Pro' badge), and a project selector (currently showing 'ejemplodeploy'). To the right of the navigation bar are links for 'Feedback', 'Changelog', 'Support', 'Docs', and a user profile icon. Below the navigation bar, there are tabs for 'Overview', 'Functions', and 'Source', with 'Overview' being the active tab. The main content area displays a project card for 'Practica web con Angular'. The card includes a preview of the application, which is a 'Crud de libros' with input fields for 'Titulo', 'Numero de paginas', and 'Autor', and a 'Guardar' button. To the right of the preview, the project's status is 'Ready' (indicated by a green dot), the environment is 'Production', and the duration is '1m 14s (3m ago)'. There is a 'Visit' button next to these details. Below the status information, the domains section shows 'ejemplodeploy.vercel.app' with a '+3' icon indicating more domains. The branch section shows 'main' with a commit hash 'a0d2379' and a 'fix' label. At the bottom right of the dashboard, there are logos for 'Santander' and 'Mision TIC 2022'.

▲ / misiontic Pro ↕ / ejemplodeploy / c1ddrsmih Feedback Changelog Support Docs

Overview Functions Source

**Practica web con Angular**  
Crud de libros

Titulo:  Categoría:   
Número de páginas:  ISBN:   
Autor:  Botón:

STATUS ● Ready ENVIRONMENT **Production** DURATION **1m 14s (3m ago)** ⋮ **Visit**

DOMAINS  
ejemplodeploy.vercel.app **+3**

BRANCH  
🔄 main  
a0d2379 — fix